HASHED SECURITY ALGORITHM FOR NON REAL TIME DATA IN MOBILE NETWORKS - SMS SECURITY

A THESIS

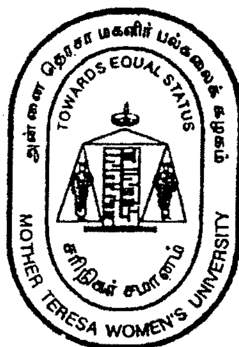
Submitted by

M.GANAGA DURGA

For the award of the degree of
DOCTOR OF PHILOSOPHY

Under the guidance of

Dr. G.CHANDRASEKARAN
Director, Department of Computer Applications
Mepco Schlenk Engineering College, Sivakasi.



DEPARTMENT OF COMPUTER SCIENCE
MOTHER TERESA WOMEN'S UNIVERSITY
KODAIKANAL
TAMIL NADU, INDIA

JANUARY 2008

CERTIFICATE

This is to certify that the thesis entitled **HASHED SECURITY ALGORITHM FOR NON REAL TIME DATA IN MOBILE NETWORKS – SMS SECURITY** submitted by **M.GANAGA DURGA** to the MOTHER TERESA WOMEN'S UNIVERSITY, KODAIKANAL for the award of the degree of DOCTOR OF PHILOSOPHY is a record of research work done by the candidate during the period of study under my supervision and that the thesis has not formed the basis for the award to the candidate of any Degree, Diploma, Associateship, Fellowship or other similar title.

Dr. G.CHANDRASEKARAN
Director,
Department of Computer Applications,
Mepco Schlenk Engineering College,
Sivakasi.

Date: 21/1/08
Place: Sivakasi

DECLARATION

This is to certify that the thesis entitled **HASHED SECURITY ALGORITHM FOR NON REAL TIME DATA IN MOBILE NETWORKS – SMS SECURITY** submitted by **M.GANAGA DURGA** to the MOTHER TERESA WOMEN'S UNIVERSITY, KODAIKANAL is a completely independent research work done by her during the period of study under my supervision and that the thesis has not formed the basis for the award to the candidate of any Degree.

Dr. G.CHANDRASEKARAN
Director,
Department of Computer Applications,
Mepco Schlenk Engineering College,
Sivakasi.

Date: 21/1/08,
Place: Sivakasi

ACKNOWLEDGEMENT

All that ends well must have a good beginning, a peculiar ado have come true with the showering of abundant blessings from yonder by the God Almighty.

For making this dream come true, I must show my gratitude in immeasurable terms to my Supervisors as it gives me immense pleasure in expressing my profound gratitude and respect to Dr. G.Chandrasekaran, Director, Department of Computer Applications, Mepco Schlenk Engineering College, Sivakasi.

I must gratefully add that his constant, dedicated support, encouragement and motivation have been the single most important reason for my completion of this research in time. Hence please permit me to supplicate myself and offer my gratitude, revered Sir!

I am also grateful to Dr. S.Arivazhagan, Professor and Head of Department of Electronics and Communication, Mepco Schlenk Engineering College, Sivakasi for his invaluable guidance and support extended to me in immeasurable terms for which I am extremely indebted. Thank you respected Sir!

I should be failing in my duty if I do not acknowledge with reverence, gratitude and thankfulness to the Principal, Mepco Schlenk Engineering College, Sivakasi who have allowed me to pursue my research under the able guidance and supervision of Dr. G.Chandrasekaran and hence please accept my deepest gratitude to you Sir.

I shall be second to none in acknowledging with respect and gratefulness to my eminent Principal, KLN College of Engineering who with fortitude have permitted me to take up the research work and who by his uncanny kindness encouraged me to compete the assignment and hence please accept my heart-felt thanks to you Sir.

Charity begins at home. I wish to express my thanks for my parents who were source of constant encouragement and valuable advice. I am indeed grateful to my beloved husband Mr. S.S. Sivakumar who with his untiring wisdom and unflinching love guided me not only with words and deeds but with deed as well and my affectionate daughter S.S.Shanmuga sunderi for their understanding and forbearance even during the time when I had to burn midnight oil and hence thank you

M. GANAGA DURGA

LIST OF TABLES

LIST OF FIGURES

CONTENTS

TOPIC **PAGE No.**

CHAPTER 1

1. INTRODUCTION

TOPIC PAGE No.

ORGANIZATION OF THE THESIS

This research work is organized with six chapters. Chapter I explains all the basic and the backbone concepts needed for this research work. Chapter II surveys the related work, and shows that the proposed scheme is new and original. Chapter III describes the proposed scheme in detail.

In Chapter IV, analysis of the tools what we have used, presented with the comparative analysis of design & vulnerability of our algorithm. In Chapter V, the simulation models and its results are presented to investigate the performance of the proposed scheme. Then the performance of the algorithm is analyzed with other algorithms also. Finally, concluding remarks and the future enhancement of this work are given in Chapter VI

CHAPTER 1

1. INTRODUCTION

1.1 WIRELESS COMMUNICATION

The term wireless is normally used to refer to any type of electrical or electronic operation which is accomplished without the use of a "hard wired" connection. Wireless communication is the transfer of information over a distance without the use of electrical conductors or "wires". The distances involved may be short (a few meters as in television remote control) or very long (thousands or even millions of kilometers for radio communications). When the context is clear the term is often simply shortened to "wireless". Wireless communications is generally considered to be a branch of telecommunications.

The term wireless technology is generally used for mobile IT (Information Technology) equipment. It encompasses cellular telephones, personal digital assistants (PDAs), and wireless networking. Other examples of wireless technology include GPS units, garage door openers and or garage doors, wireless computer mice and keyboards, satellite television and cordless telephones.

Wireless operations permits services, such as long range communications, that are impossible or impractical to implement with the use of wires. The term is commonly used in the telecommunications industry to refer to telecommunications systems (e.g., radio transmitters and receivers, remote controls, computer networks, network terminals, etc.) which use some form of energy (e.g. radio frequency (RF), infrared light, laser light, visible light, acoustic energy, etc.) to transfer information without the use of wires.

Information is transferred in this manner over both short and long distances. Wireless communication may be via: radio frequency communication, microwave communication, for example long-range line-of-sight via highly directional antennas, or short-range communication, or infrared (IR) short-range communication, for example from remote controls or via IRDA. Applications may involve point-to-point communication, point-to-multipoint communication, broadcasting, cellular networks and other wireless networks.

Discussions on different multiple-access schemes can be found in all classic communication texts such as References [1–4] and in more specialized literature [5–13]. Frequency modulation (FM) gave the idea of sharing the frequency among many people so that they send their signals at different frequency bands.

The term "wireless" should not be confused with the term "cordless", which is generally used to refer to powered electrical or electronic devices that are able to operate from a portable power source (e.g., a battery pack) without any cable or cord to limit the mobility of the cordless device through a connection to the main power supply.

Some cordless devices, such as cordless telephones, are also wireless in the sense that information is transferred from the cordless telephone to the telephone's base unit via some type of wireless communications link. This has caused some disparity in the usage of the term "cordless", for example in Digital Enhanced Cordless Telecommunications. In the last 50 years, wireless communications industry experienced drastic changes driven by many technology innovations.

1.1.1. The Wireless Revolution

Executives and professional field forces are spending more time on the road doing business. These mobile professionals must be readily accessible to customers, partners and co-workers. In the past, this required that they carry laptops and use cumbersome and expensive remote-access systems such as Virtual Private Networks (VPN). Today, advances in handheld and network technology mean that laptops are no longer needed for secure wireless access to e-mail and other mission-critical enterprise systems.

No longer just a luxury for top executives, mobile technology has become a necessity for field forces. Mobile access to enterprise information systems drives productivity and efficiency. Handheld mobile applications are changing the way that companies, employees and customers conduct business. These technologies can improve business processes in sales, service, marketing and logistics such as radio transmitters, wireless communications systems, and the like, were base stations, operated at fixed locations, typically with large antenna towers.

Widespread use of automobiles gave rise to smaller devices operating at 6 volts. In the 1950s, the transition to 12 volt automotive electrical systems gave rise to a large number of 12 volt devices, such as two-way radios, referred to as mobile rigs. A large industry, with companies such as Motorola sprung up to support the growing need for mobile devices, such as taxicab radios, police radios, and other 12 volt under dash equipment, as well as trunk mount systems. Today there are a wide variety of mobile computing platforms, including dash-mount VGA displays, and computers that can provide GPS and other navigation functions for automobile user.

The term "wireless" came into public use to refer to a radio receiver or transceiver (a dual purpose receiver and transmitter device), establishing its usage in the field of wireless telegraphy early on; now the term is used to describe modern wireless connections such as in cellular networks and wireless broadband Internet.

It is also used in a general sense to refer to any type of operation that is implemented without the use of wires, such as "wireless remote control", "wireless energy transfer", etc. regardless of the specific technology (e.g., radio, infrared, ultrasonic, etc.) that is used to accomplish the operation.

Wireless is a term used to describe telecommunications in which electromagnetic waves (rather than some form of wire) carry the signal over part or the entire communication path. Common examples of wireless equipment in use today include:

Cellular phones and pagers: provide connectivity for portable and mobile applications, both personal and business. Cordless telephone sets are limited-range devices, not to be confused with cell phones.

Global Positioning System (GPS): allows drivers of cars and trucks, captains of boats and ships, and pilots of aircraft to ascertain their location anywhere on earth.

Cordless computer peripherals: the cordless mouse is a common example; keyboards and printers can also be linked to a computer via wireless.

Satellite television: allows viewers in almost any location to select from hundreds of channels.

Wireless networking is used to meet a variety of needs. Perhaps the most common use is to connect laptop users who travel from location to location. Another common use is for mobile networks that connect via satellite. A wireless transmission method is a logical choice to network a LAN segment that must frequently change locations.

The following situations justify the use of wireless technology:

- To span a distance beyond the capabilities of typical cabling,
- To avoid obstacles such as physical structures, EMI (Electromagnetic Interference), or RFI (Radio Frequency Interference),
- To provide a backup communications link in case of normal network failure,

- To link portable or temporary workstations,
- To overcome situations where normal cabling is difficult or financially impractical, or
- To remotely connect mobile users or networks.

1.1.2. The Essential Elements of Wireless Security

Maintaining security while providing mobile workers with access to the information they need when and where they need it is complex. Protecting enterprise IT infrastructure requires a deep understanding of the risks associated with mobile applications, handhelds and wireless networks.

The move toward wireless data access extends the perimeter of the corporate network and, like earlier innovations, raises many security issues. Compared with behind-the-firewall enterprise systems, wireless handheld computing systems are fundamentally different and involve incremental security risks.

Perimeter or firewall security—when a corporation wishes to make enterprise systems like enterprise messaging servers, CRM, ERP or intranet Web pages accessible wirelessly, the first priority is to maintain the security of the internal network. Any programs running inside the firewall must not open avenues of attack from programs running outside.

Additional perimeter security considerations include:

Authentication—each component of a wireless system must be able to prove that it is authorized to communicate on the network. It must not be possible for an attacker to impersonate a handheld or server, thereby misleading authentic services into communicating with it.

Administrative security—Enterprises need to ensure that different administrative tasks are accessible only to the appropriate administrator. For example, only the most senior system administrators may modify system-wide security policies while lower level administrators may provision new users.

Transmission/over-the-air (ota) security—when internal information is transmitted over the public Internet and/or a wireless network, the data must be protected against interception or "man-in-the-middle" attacks. Data packets can be intercepted and read if unencrypted or weakly encrypted transmission security is employed. The handheld session itself can be hijacked and an unauthorized user can interact with backend systems if transmission and authentication security is not robust.

Handheld security—Once internal information is received and decrypted for viewing on a handheld, that information must be protected against access by unauthorized users or programs on the handheld. Handheld security must also address corporate requirements to control various functions on the handheld (like use of Wi-Fi, Bluetooth, cameras, speakers, etc.) as well as provide IT managers with a mechanism to control which applications are used on a handheld.

1.1.3. Security Issues in Wireless Networks

Recent years have seen an explosive growth of interest in wireless networks that support the mobility of users (and terminals). These networks serve as a foundation of future universal, mobile and ubiquitous personal communications systems.

Emerging wireless networks share many common characteristics with traditional wire-line networks such as public switched telephone/data networks, and hence many security issues with wire-line networks also apply to the wireless environment.

Nevertheless, the mobility of users, the transmission of signals through open-air and the requirement of low power consumption by a mobile user bring to a wireless network with a large number of features distinctively different from those seen in a wire-line network.

Especially, security and privacy becomes more eminent with wireless networks [14]. To this end, the primarily concern is with security issues related to or caused by the mobility of users/terminals, open-air transmission of signals and low power supply of a mobile user.

When examining security in a wireless network, ranges of issues have to be taken into account [15]. These issues include:

- Identification of a mobile user
- anonymity of a mobile user (protection of identity)
- authentication of a base station
- security of information flowing between a mobile user and a base station
- prevention of attacks from within a base station
- hand-over of authentication information
- The communication cost of establishing a session key between a mobile user and a base station, which is indicated primarily by the total number and length of message to be exchanged.
- The cost of communications between a mobile user's home domain and a foreign domain where he is currently located, as well as

security requirements on the communication links between the two domains.

- The computational complexity of achieving authenticity and security.
- The complexity of computations to be carried out by a mobile user's terminal, which is in general much less powerful than a base station.

1.1.4. Summary

Wireless communication is the root for our research work. Hence the communication of wireless devices and its networks with all its revolutions discussed here. Next, the heart of our research is the security therefore the elements of security and its issues to be addressed are presented.

1.2. MOBILE COMPUTING

Whilst Scandinavia is leading the world in the penetration of mobile phones, the UK is rapidly catching up since the introduction of pre-paid mobiles which currently account for 80% of new sales. Within the UK there are 4 major network operators: BT Cell net, One2One, Orange and Vodafone. As a whole, it is believed that Europe is ahead of the USA thanks to the standardization on GSM digital mobiles.

Never-the-less, in this highly capital-intensive business size is everything, and global players are beginning to form. Deutsche Telekom purchased One2One, Vodafone purchased the US AirTouch for £38bn, and Mannesmann bought Orange for £20bn. Now Vodafone AirTouch is in a hostile £80bn takeover of Mannesmann which would necessitate the spin-off of Orange.

Impressive though these figures are, we are just at the start of the mobile revolution as use shifts from voice calls to data calls and soon data and voice will be mixed in the same transaction [17]. Analysts predicted that between 2002 and 2005 consumers and business will purchase 600m internet enabled mobile phones. Durlacher has predicted that the European m-commerce market (that's mobile e-commerce including value added services) will be worth Ecu 2bn (£14.5bn) by 2003 from just Ecu 323m (£200m) now.

Already there is a shift to acquiring new customers for potential m-commerce revenue rather than current voice revenue. This could bring voice call down to less than BT's fixed line charges unless BT also gives us (near) free calls in return for purchasing other services like video-on-demand. Besides the mobile phone, many other devices are being enabled for communications [18]. Personal Digital Assistants (PDAs) have become very popular consumer items with many users using them to support their work and most recently the worlds first internet enabled washing machine has been put on sale.

All these devices will pave the way for applications that provide added value to the utility of plain voice calls. Short Message Text services are already common place.

With the recent launch of WAP (Wireless Application Protocol) enabled mobile phones it is now only a matter of days before web based applications are made available to the mobile community. For the future, UMTS (Universal Mobile Telecommunications System) phones are being prototyped which will provide high speed data transmission opening the way to more demanding applications including video.

Financial service providers will need to embrace and adapt to the mobile mediums. It will change the way they communicate to their prospects and customers, how they respond to purchase requests, and how they service enquires and claims. In essence, it will be a far more interactive and dynamic world, where the concept of any time, any way, and any how comes to fruition.

Mobile Computing is a generic term describing our ability to use technology 'untethered', that is not physically connected, or in remote or mobile (non static) environments. The term is evolved in modern usage such that it requires that the mobile computing activity be connected wirelessly to and through the internet or to and through a private network. This connection ties the mobile device to centrally located information

and/or application software through the use of battery powered, portable, and wireless computing and communication devices [16].

This includes devices like laptops with wireless LAN or wireless WAN technology, smart mobile phones, wearable computers and Personal Digital Assistants (PDAs) with Bluetooth or IRDA interfaces.

Many types of mobile computers have been introduced since the 1990s, including:

- Laptop computer
- Sub notebook
- Personal digital assistant (PDA)
- Portable data terminal (PDT)
- Mobile data terminal (MDT)
- Tablet personal computer
- Smartphone
- Ultra Mobile Personal Computer (UMPC)

1.2.1. Evolution of Mobile Cellular Networks

The first generation (1G) and second generation (2G) of mobile telephony were intended primarily for voice transmission. The third generation of mobile telephony (3G) will serve both voice and data applications.

3G–i.e., an entirely packet switched network with all digital network elements and extremely high available bandwidth [19]. For the most part, it is believed that 4G will bring true multimedia capabilities such as high-speed data access and video conferencing to the handset.

It is also envisioned that 4G systems will be deployed with software defined radios, allowing the equipment to be upgraded to new protocols and services via software upgrades. 4G also holds the promise of worldwide roaming using a single handheld device.

The original analog cellular systems are considered as the first generation of mobile telephony (1G). In the early 1980s, 1G system was deployed. At the same time, the cellular industry began developing the second generation of mobile telephony (2G). The difference between 1G and 2G is in the signaling techniques used: 1G used analog signaling, 2G used digital signaling.

As experience shows, the lead-time for mobile phone systems development is about 10 years. It was not until the early to mid 1990s that 2G was deployed. Primary thinking and concept development on 3G generally began around 1991 as 2G systems just started to roll out. Since the general model of 10 years to develop a new mobile system is being followed, that timeline would suggest 4G should be operational some time around 2011. 4G would build on the second phase of 3G, when all networks are expected to embrace Internet protocol (IP) technology [20].

2.5G is the interim solution for current 2G networks to have 3G functionality. 2.5G networks are being designed such that a smooth transition (software upgrade) to 3G can be realized [21]. 2.5G networks currently offer true data speeds up to 28kbps.

In comparison, the theoretical speed of 3G can be up to 2 Mbps [22], i.e., approximately 200 times faster than previous 2G networks. It is anticipated that 4G speeds could be as high as 100 Mbps.

Mobile Internet services have been started in the second phase of the second-generation (2G) cellular networks, such as the General Packet Radio Service (GPRS) [23–25], the successor of the worldwide Global System for Mobile communications (GSM) cellular network [26].

The current network architectures used in either the wired Internet or the cellular networks would not be appropriate and efficient for future wireless mobile Internet, even if the cellular networks will provide the major infrastructure of the mobile Internet. In recent years, many literatures have discussed this issue and how it is possible to change the network architecture to be utilized for the mobile Internet [27–33].

First Generation Mobile Systems: The first generation of analog cellular systems included the Advanced Mobile Telephone System (AMPS) which was made available in 1983. A total of 40MHz of spectrum was allocated from the 800MHz band by the Federal Communications Commission (FCC) for AMPS.

It was first deployed in Chicago, with a service area of 2100 square miles. AMPS offered 832 channels, with a data rate of 10 kbps. Although omni directional antennas were used in the earlier AMPS implementation, it was realized that using directional antennas would yield better cell reuse.

In fact, the smallest reuse factor that would fulfill the 18db signal-to-interference ratio (SIR) using 120-degree directional antennas was found to be 7. Hence, a 7-cell reuse pattern was adopted for AMPS. Transmissions from the base stations to mobiles occur over the forward channel using frequencies between 869-894 MHz. The reverse channel is used for transmissions from mobiles to base station, using frequencies between 824-849 MHz.

In Europe, TACS (Total Access Communications System) was introduced with 1000 channels and a data rate of 8 kbps. AMPS and TACS use the frequency modulation (FM) technique for radio transmission. Traffic is multiplexed onto an FDMA (frequency division multiple access) system. In Scandinavian countries, the Nordic Mobile Telephone is used.

Second-Generation Mobile Systems: Compared to first-generation systems, second-generation (2G) systems use digital multiple access technology, such as TDMA (time division multiple access) and CDMA (code division multiple access). Global System for Mobile Communications, or GSM, uses TDMA technology to support multiple users.

Examples of second-generation systems are GSM, Cordless Telephone (CT2), Personal Access Communications Systems (PACS), and Digital European Cordless Telephone (DECT). A new design was introduced into the mobile switching center of second-generation systems.

In particular, the use of base station controllers (BSCs) lightens the load placed on the MSC (mobile switching center) found in first-generation systems. This design allows the interface between the MSC and BSC to be standardized. Hence, considerable attention was devoted to interoperability and standardization in second-generation systems so that carrier could employ different manufacturers for the MSC and BSCs.

In addition to enhancements in MSC design, the mobile-assisted handoff mechanism was introduced. By sensing signals received from adjacent base stations, a mobile unit can trigger a handoff by performing explicit signaling with the network.

Second generation protocols use digital encoding and include GSM, D-AMPS (TDMA) and CDMA (IS-95). 2G networks are in current use around the world. The protocols behind 2G networks support voice and some limited data communications, such as Fax and short messaging service (SMS), and most 2G protocols offer different levels of encryption and security. While first-generation systems support primarily voice traffic, second-generation systems support voice, paging, data, and fax services.

2.5G Mobile Systems: The move into the 2.5G world will begin with General Packet Radio Service (GPRS). GPRS is a radio technology for GSM networks that adds packet-switching protocols, shorter setup time for ISP connections, and the possibility to charge by the amount of data sent, rather than connection time. Packet switching is a technique whereby the information (voice or data) to be sent is broken up into packets, of at most a few Kbytes each, which are then routed by the network between different destinations based on addressing data within each packet.

Use of network resources is optimized as the resources are needed only during the handling of each packet. The next generation of data heading towards third generation and personal multimedia environments builds on GPRS and is known as Enhanced Data rate for GSM Evolution (EDGE). EDGE will also be a significant contributor in 2.5G.

It will allow GSM operators to use existing GSM radio bands to offer wireless multimedia IP-based services and applications at theoretical maximum speeds of 384 kbps with a bit-rate of 48 kbps per timeslot and up to 69.2 kbps per timeslot in good radio conditions.

EDGE will let operators function without a 3G license and compete with 3G networks offering similar data services. Implementing EDGE will be relatively painless and will require relatively small changes to network hardware and software as it uses the same TDMA (Time Division Multiple Access) frame structure, logic channel and 200 kHz carrier bandwidth as today's GSM networks.

As EDGE progresses to coexistence with 3G WCDMA, data rates of up to ATM-like speeds of 2 Mbps could be available. GPRS will support flexible data transmission rates as well as continuous connection to the network. GPRS is the most significant step towards 3G.

Third-Generation Mobile Systems: Third-generation mobile systems are faced with several challenging technical issues, such as the provision of seamless services across both wired and wireless networks and universal mobility. In Europe, there are three evolving networks under investigation: (a) UMTS (Universal Mobile Telecommunications Systems), (b) MBS (Mobile Broadband Systems), and (c) WLAN (Wireless Local Area Networks).

The use of hierarchical cell structures is proposed for IMT2000. The overlaying of cell structures allows different rates of mobility to be serviced and handled by different cells. Advanced multiple access techniques are also being investigated, and two promising proposals have evolved, one based on wideband CDMA and another that uses a hybrid TDMA/CDMA/FDMA approach.

1.2.2. History of Mobile Devices

Originally, electronic devices such as radio transmitters, wireless communications systems, and the like, were base stations, operated at fixed locations, typically with large antenna towers. A large industry, with companies such as Motorola sprung up to support the growing need for mobile devices, such as taxicab radios, police radios, and other 12 volt underdash equipment, as well as trunk mount systems.

Today there are a wide variety of mobile computing platforms, including dash-mount VGA displays, and computers that can provide GPS and other navigation functions for automobile users. Mobile computing is a technology that allows transmission of data, via a computer, without having to be connected to a fixed physical link.

The term is evolved in modern usage such that it requires that the mobile computing activity be connected wirelessly to and through the internet or to and through a private network. This connection ties the mobile device to centrally located information and/or application software through the use of battery powered, portable, and wireless computing and communication devices.

It is also known as being able to use a computing device even when being mobile and therefore changing location. Portability is one aspect of mobile computing. This includes devices like laptops with wireless LAN technology, mobile phones, wearable computers and Personal Digital Assistants (PDAs) with Bluetooth or IRDA interfaces, and USB flash drives [34]. Mobile voice communication is widely established throughout the world and has had a very rapid increase in the number of subscribers to the various cellular networks over the last few years.

An extension of this technology is the ability to send and receive data across these cellular networks. This is the principle of mobile computing. Mobile data communication has become a very important and rapidly evolving technology as it allows users to transmit data from remote locations to other remote or fixed locations. This proves to be the solution to the biggest problem of business people on the move - mobility.

1.2.3. Summary

This chapter helps us to differentiate wireless network with mobile or cellular network. The mobile devices and its communications methodologies are presented here. Then all about the mobile evolutions with their working principles are also explained.

1.3. MOBILE TECHNOLOGIES

There is a lot of hype in the technology industry about "mobile" and "wireless" computing solutions but very little in the way of basic information about what it is and why it's important. It's no surprise that when technophiles toss around terms such as "fixed success points", “network latency" and "2.5+ digital wireless" , most ordinary folk roll over and go back to sleep.

The first thing to know is that when two devices connect or "talk" to each other without a physical cable between the two, they use radio frequencies to transmit information. This is important because different vendors use different frequencies, cover different areas and have a wide range of signal strengths.

The available mobile technologies include IEEE802.11 Wireless LAN, Blue tooth radio system, Multi-hop wireless LAN, GPRS, 3G and satellite wireless systems and the access methods suggested by also include WAP, AvantGo, and VoiceXML. Among the above technologies, WAP is the first feasible solution. WAP (Wireless Application Protocol) is a protocol designed for easy fast delivery of relevant information and services to mobile users of handheld digital wireless devices such as mobile phones, pagers, two-way radios, smartphones, communicators and PDAs.

WAP is a communication protocol and an application environment. It can be built on any operating system including PalmOS, EPOC, Windows CE, FLEXOS, OS/9, JavaOS and so on. WAP-enabled mobile devices can connect to the Internet via a WAP gateway. The WAP gateway connects the mobile network and the Internet by translating the Hypertext Transfer Protocol (HTTP) to the Wireless Session Protocol (WSP). The content on the Internet are also converted from HTML format to WML format by the WAP gateway.

The limitations are: WAP pages should be less than 1400 octets in size to assure general operability and bandwidth is restricted to 9600 bps. However such limitation can be overcome through proper interface design by keeping some usability principles for mobile Internet applications:

1. Less is more;
2. The user is mobile;
3. Keep in mind the display size;
4. Navigation;
5. Designing for fixed and mobile Internet.

Generally, the technical standard for Mobile Internet is the WAP protocol stack, with XHTML as the markup language, and either a proprietary operating system (OS) or some standard platform like Symbian. Moreover, it should be "device independent". The WAP services include news, weather forecast, album, mail, game, chat, study and so on.

Another approach for developing mobile applications is Java 2 Platform, Micro Edition (J2ME) J2ME can be used either as a complementary technology for WAP, cHTML (compact HTML), and i-Mode, or as standalone J2ME applications on mobile devices.

What J2ME can do for mobile devices is just the same as what Java 2 Standard Edition (J2SE) and Java 2 Enterprise Edition (J2EE) did for desktop and server systems. An example of J2ME application is the wireless online virtual community, MOOsburg++ .

The third possible solution is the AvantGo M-Business Server. AvantGo M-Business Server is designed to deliver the web to mobile devices —extending HTML-based resources simply and efficiently to mobile users. We can create handheld-friendly "channels" of information in hours or days, instead of months, using standard web-based tools, such as Macromedia Dream weaver or Microsoft FrontPage.

AvantGo's software is also uniquely device-agnostic; learners are able to use a variety of devices, including Windows Powered Pocket PCs and Palm OS handhelds. The mobile devices and mobile networks have some limitations compared to the desktop systems: The bandwidth of wireless networks is quite low, around 9.6 Kbps; the screen size is very small, and even not colored; the CPU and memory capacity are both limited with limited input facilities. These limitations have overcome by the mobile technologies.

1.3.1. Wireless Application Protocol [WAP]

WAP tends to solve this bandwidth issue by using binary encoding to minimize the traffic [35]. WML and WMLscript are binary encoded into a compact form before they are transmitted. The Wireless Session Protocol (WSP), which is equivalent to HTTP on the Internet, is also binary encoded for the same reason.

In addition, WSP supports both sessions that can be suspended and resumed, and header caching. WML structures its document in decks and cards. A card is a single unit of interaction with the end user and is small enough to be displayed on a small screen. WAP handles the limited CPU and memory by defining a lightweight protocol stack.

The limited set of functionalities provided by WML and WMLscript make it, possible to implement browsers that require less computation power and ROM resources. The binary encoding and WMLscript helps to minimize the use of RAM. The elements used in WML can be easily implemented so that they seldom use a keyboard.

The use of decks and cards makes the users to navigate a series of cards instead of scroll up and down on one large page, which also reduces the use of keyboard. Soft-button, or user-definable keys, are supported by WML. Users can achieve certain functions by just one click.

1.3.2. J2ME (Java Micro Edition)

J2ME uses Connected Limited Device Configuration [CLDC] and Mobile Information Device Profile (MIDP) for resource constrained mobile devices. A configuration defines a Java platform for a "horizontal" category or grouping of devices with similar requirements on total memory budget and other hardware capabilities.

CLDC specifies all features supported by the Java programming language, the Java virtual machine, the basic Java libraries and APIs. The paper [36] specifies the necessary requirements for devices that can run under J2ME CLDC which are: wireless, intermittent connection with limited (often 9600 bps or less) bandwidth; at least 160 kilobytes of total memory; processor speed starting from 8 to 32 Mhz.

The MIDP is designed to be used with the CLDC. A profile is a set of APIs that reside on top of a configuration that offers the program access to device-specific capabilities. MIDP provides a set of APIs for use by mobile devices. It contains classes for user interface, persistence storage and networking. It also includes a standardized runtime environment that allows new applications to be downloaded to end user devices.

In addition to device requirements of CLDC, the MIDP defined a Mobile Information Device (MID) to be a device that has a screen size of 96*54 and an input facility of one-handed keypad, two-handed keyboard or touch screen. CLDC and MIDP commonly run on top of Sun's K Virtual Machine (KVM). KVM is also designed for small, resource-constrained devices. The KVM is a complete Java runtime environment for small devices. It is suitable for 16/32 bit microprocessors with a total memory of at least 128 kilobytes.

Actually WAP/WML and Java can work together to meet the needs from different mobile device users, for example, Java servlets and Java Server Pages used to generate WML pages dynamically.

1.3.3. AvantGO

AvantGo is the most widely-used channel system available for all types of PDA devices regardless of their operating systems (WinCE, pocketPC, PalmOS, Epoc, etc.) [37]. It serves over 3.5 million users. Once registered in the system users download and install the software to their handheld computer. This enables them to choose from thousands of different channels to which they can subscribe worldwide. Most channels are free but some, such as business-type channels, require payment of a small fee.

All users have to do is to synchronize PDA devices with the AvantGo server, which then displays current content from the provider. This can be done either via a PC (using cradle-link) or wireless (if the device is able to communicate independently or via a mobile phone). The system is usually set up by the user to automatically update the channel at least once in a day.

1.3.4. Summary

This section exposes all the mobile technologies to implement real time mobile based applications and points out the best mobile technology used to create globalized application.

1.4. CRYPTOGRAPHY

The origin of the word cryptology lies in ancient Greek. The word cryptology is made up of two components: "kryptos", which means hidden and "logos" which means word. Cryptology is as old as writing itself, and has been used for thousands of years to safeguard military and diplomatic communications. For example, the famous Roman emperor Julius Caesar used a cipher to protect the messages to his troops.

Within the field of cryptology one can see two separate divisions: cryptography and cryptanalysis. The cryptographer seeks methods to ensure the safety and security of conversations while the cryptanalyst tries to undo the formers work by breaking his systems.

Cryptography is the science of using mathematics to encrypt and decrypt data. Cryptography enables us to store sensitive information or transmit it across insecure networks so that it cannot be read by anyone except the intended recipient. While cryptography is the science of securing data, cryptanalysis is the science of analyzing and breaking secure communication. Classical cryptanalysis involves an interesting combination of analytical reasoning, application of mathematical tools, pattern finding, patience, determination, and luck. Cryptanalysts are also called attackers. Cryptology embraces both cryptography and cryptanalysis.

1.4.1. Encryption and Decryption

Data that can be read and understood without any special measures is called plaintext or clear text. The method of disguising plaintext in such a way as to hide its substance is called encryption. Encrypting plaintext results in unreadable gibberish called cipher text. Encryption is used to ensure that information is hidden from anyone for whom it is not intended, even those who can see the encrypted data. The process of reverting cipher text to its original plaintext is called decryption.

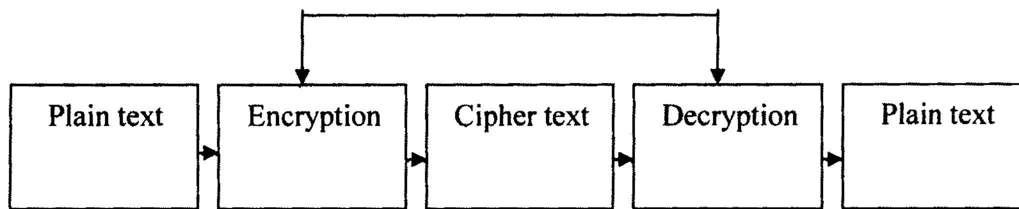


Figure.1: Cryptology steps

Working Principle of Cryptography: A cryptographic algorithm, or cipher, is a mathematical function used in the encryption and decryption process. A cryptographic algorithm works in combination with a key — a word, number, or phrase — to encrypt the plaintext.

The same plaintext encrypts to different cipher text with different keys. The security of encrypted data is entirely dependent on two things: the strength of the cryptographic algorithm and the secrecy of the key. A cryptographic algorithm, plus all possible keys and all the protocols that make it work comprise a cryptosystem.

1.4.2. Types of Cryptography

The cryptography is divided into conventional and public key cryptography. The first one is the private key usage to access but the latter uses the public key followed by private key.

Conventional Cryptography: In conventional cryptography, also called secret-key or symmetric-key encryption, one key is used both for encryption and decryption [38]. The Data Encryption Standard (DES) is an example of a conventional cryptosystem that is widely employed by the Federal Government.

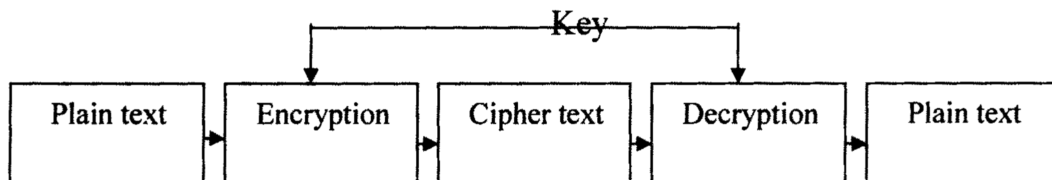


Figure: 2. Conventional cryptosystem

Public Key Cryptography: Public key cryptography is an asymmetric scheme that uses a pair of keys for encryption: a public key, which encrypts data, and a corresponding private, or secret key for decryption. The public key is published to the world while the private key is kept as secret key [39].

1.4.3. Summary

Encryption and decryption used to provide privacy and security sheath for our data to send or receive in any type of networks. This chapter explains the same in detail.

1.5. DATA CLASSIFICATION

There are two types of traffic in wireless networks:

1. Real-time traffic - video and voice traffic from users
2) Non real-time traffic – non real-time data traffic such as SMS, E-mail, Instant messages and other TCP/IP traffic.

1.6. SECURITY THREATS

Theft, fraud and forgery have become a major threat to private individuals as well as the world's economies. Industrial espionage results in damage worth billions of Euros per year. With the spread of mobile phones and internet usage has raised the danger for individuals, companies and governmental institutions dramatically. Hackers can phone on a hijacked mobile phone account. Wiretapping, eavesdropping, and identity theft are among the more obvious dangers, but more threats that are secretive exist as well.

Mobile computing presents many new opportunities for application developers, but these opportunities also come at the expense of increased security threats. These threats exist due to limitations in mobile networks, software, and hardware.

1.6.1. E-Mail Security Threats

In the computer era, so many threats to the data. In which the threats to E-mail is very much dangerous which in turn spoils the system or entire network.

Viruses: Email security is threatened by a range of issues. One of the most publicized and high risk of all the issues is viruses. Viruses are so dangerous because they often deliver extremely destructive payloads, destroying data, and bringing down entire mail systems. As a result they are a major drain on corporate IT departments and users.

SPAM: Another major threat to email security today is SPAM, often cited by organizations as being their number one concern. Otherwise known as junk email [40], SPAM is considered as a security threat not only because the volume of it can affect system availability, but also because it can carry viruses, malicious code, and fraudulent solicitations for private information. It is an ever-growing problem that is of particular concern to information security.

Phishing : Phishing, also known as identify theft, is a newer threat to email security that was relatively unheard of one year ago. Phishing is the process whereby identity thieves target customers of financial institutions and high-profile online retailers, using common spamming techniques to generate large numbers of emails with the intent of luring customers to spoofed web sites and tricking them into giving up personal information such as passwords and credit card numbers.

1.6.2. Threats to Instant Messaging

Threats to instant messengers are not limited to worms, but also include Trojan horses that export data and create back doors into the system. Furthermore, one of the greatest threats of utilizing any instant messenger is simply privacy.

Related to IM security, a modified Diffie-Hellman protocol suitable to instant messaging has been designed by Kikuchi etal. [41] Primarily intended to secure message confidentiality against IM servers. It does not ensure authentication and also has problems similar to the IMSecure3. Hindocha [42] discusses popular IM protocols, worms, threats and firewall issues in a 2003 white paper.

A Web resource on security analysis of Cerulean Studios' Trillian application is also available [43]. Informal discussions of security problems related to public instant messaging in the enterprise environment are available in this paper [44]. As none of the popular instant messaging service protects their connections with encryption, it is quite easy to impersonate any connection via man-in-the-middle attacks [42].

Worms: The number of instant messaging worms is rising steadily, but there are still no antivirus applications that directly monitor instant messaging traffic and only a few that directly plug in to instant messaging clients, being notified when a file is received. This is partly due to the difficulty in monitoring instant messaging traffic, as well as the constant modifications to the clients and the protocols that they use. Unfortunately, this makes instant messengers an open door to the computer, as the traffic will pass most server-based security measures unstained for potential worms.

Backdoor Trojan Horses: One can share every file on a person's computer using an instant messenger. All the popular instant messengers have file sharing capabilities, or the ability to add such functionality by applying patches or plug-ins.

The benefit for a hacker using an instant messenger to access files on a remote computer. Instead of installing a backdoor Trojan horse is that even if the computer is using a dynamic IP address, the screen name will probably never change.

Furthermore, the hacker will receive a notification each time the victim computer is online. This will make it much easier for the hacker to keep track of and access infected computers. Backdoor Trojan horses that allow file-access to the computer by utilizing instant messenger clients may be harder to discover than classic backdoor Trojan horses. Classic backdoor Trojan horses open a listening or outgoing port on the computer, forming a connection with a remote machine.

Hijacking and Impersonation: There are many different ways in which hackers can impersonate other users. The most frequently used attack is simply stealing the account information of an unsuspecting user. To get the account information of a user, the hacker can use a password-stealing Trojan horse.

Denial of service: There are many ways in which a hacker can cause a denial of service on an instant messenger client. Some denials of service attacks make the instant messaging client crash. Other types of attacks will make the client hang, and in some cases consume a large amount of CPU power, causing the entire computer to become unstable.

One common type of attack is flooding a particular user with a large number of messages. The various instant messaging clients do contain a protection against flood-attacks by allowing the victim to ignore certain users.

Information Disclosure: Tools that attempt to retrieve the system information from instant messenger users are in very common use today. An example of such a tool is an IP address retriever. If an IP address retriever was used together with a backdoor Trojan horse, the hacker could receive a message containing the IP address of an infected user each time the victim comes online this way, the hacker would know the IP address of the infected user, even if the user were using dynamic IP addresses.

1.6.3. Summary

In this section, the threats to e-mail, mobile chat messages and how it is hacked or affected by unauthorized persons are elaborately defined.

1.7. SHORT MESSAGE SERVICE [SMS]

Mobile usage is increasing in volume as well as diversity as the mobile phone is already an integral part of the lives of more than 1 billion people worldwide. More than 80 % of mobile users do not leave home without their phones. Businesses are increasingly turning to the mobile phone to "get the message across" to the employees anywhere anytime. The desire to communicate more easily and have more timely access to information is universal.

The mobile phone is today being adopted in innovative ways to enhance business productivity and SMS is playing a leading role in this adoption. SMS is a widely used service for brief communication. Occasionally the data sent using SMS services is confidential in nature and is desired not to be disclosed to a third party.

SMS stands for Short Message Service. It is a technology that enables the sending and receiving of messages between mobile phones. SMS first appeared in Europe in 1992. It was included in the GSM (Global System for Mobile Communications) standards right at the beginning [45]. Later it was ported to wireless technologies like CDMA and TDMA.

The GSM and SMS standards were originally developed by ETSI (European Telecommunications Standards Institute). Now the 3GPP (Third Generation Partnership Project) is responsible for the development and maintenance of the GSM and SMS standards.

As suggested by the name "Short Message Service", the data that can be held by an SMS message is very limited. One SMS message can contain at most 140 bytes (1120 bits) of data, so one SMS message can contain up to [46]:

- 160 characters if 7-bit character encoding is used. (7-bit character encoding is suitable for encoding Latin characters like English alphabets.)
- 70 characters if 16-bit Unicode UCS2 character encoding is used. (SMS text messages containing non-Latin characters like Chinese characters should use 16-bit character encoding.)
- 450 characters for the high end phones.

SMS text messaging supports languages internationally. It works fine with all languages supported by Unicode, including Arabic, Chinese, Japanese and Korean. Besides text, SMS messages can also carry binary data. It is possible to send ring tones, pictures, operator logos, wallpapers, animations, business cards (e.g. VCards) and WAP configurations to a mobile phone with SMS messages.

One major advantage of SMS is that it is supported by 100% GSM mobile phones. Almost all subscription plans provided by wireless carriers include inexpensive SMS messaging service. Unlike SMS, mobile technologies such as WAP and mobile Java are not supported on many old mobile phone models.

SMS is having enormous popularity as an economical and convenient mode of exchanging information. It not only saves time and cost but in many situations SMS is found to be more convenient than talking on the phone. SMS has changed our working habits and social lives in many ways. SMS has simplified exchange of important short messages and also lead to creation of services that are just fun to use.

People can easily share a private moment with their friends, family and work in other geographical sites in a cost effective and instant manner. SMS is further being used in business tasks such as simplifying grocery shopping, giving alert of a best buy or any monitored event [47].

Besides that it is being used these days in getting daily NEWS, stocks information, sports scores, quotes, travel and weather news. Many value added services such as contest voting, songs request, ring tone or service initiation is being also done using SMS. There are so many services that are being churned out because of widespread acceptance of SMS that it just cannot be summarized.

1.7.1. SMS Architecture

Short messages are delivered in GSM signaling channels between the MS (Mobile station) and the BSS (Basic service station). SMS network architecture and its operations, as well as other scenarios with fixed entities that are capable of sending and receiving short messages [48] and its specification [49] are discussed.

The messages flow as normal calls, but they are routed from the MSC to a short message service center (SMSC). The SMSC stores the message until it can be delivered to the recipient(s) or until the message's validity time has elapsed. The recipient can be a normal MS user or a SMS gateway. The gateways are servers, which are connected to one or more SMSC's to provide SMS applications for the MS users. These applications include ring tone and icon delivery, entertainment, bank services, and many other beneficial services.

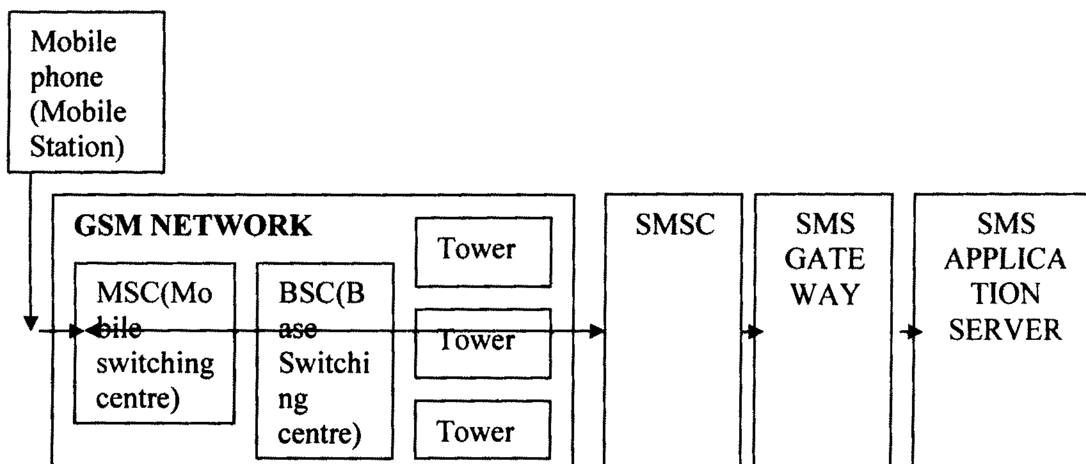


Figure.3 Architecture of SMS

1.7.2. SMS Working Principle

The transmission of short text messages to and from a mobile phone, fax machine and/or IP address. Messages must be no longer, than 160 alpha-numeric characters and contain no images or graphics. Once a message is sent, it is received by a Short Message Service Center (SMSC), which must then get it to the appropriate mobile device.

To do this, the SMSC sends a SMS Request to the home location register (HLR) to find the roaming customer. Once the HLR receives the request, it will respond to the SMSC with the subscriber's status: 1) inactive or active 2) where subscriber is roaming.

If the response is "inactive", then the SMSC will hold onto the message for a period of time. When the subscriber accesses his device, the HLR sends a SMS Notification to the SMSC, and the SMSC will attempt delivery [50]. The SMSC transfers the message in a Short Message Delivery Point to Point format to the serving system.

The system pages the device, and if it responds, the message gets delivered. The SMSC receives verification that the message was received by the end user, then categorizes the message as "sent" and will not attempt to send again.

1.7.3. Types of SMS

Smart Messaging: Smart Messaging was developed by Nokia. Smart Messaging allows Nokia phone users to create messages that have pictures, ring tones, virtual business cards and other types of non-text data, all within a message that is roughly compatible with the SMS text standard. In other words, although these messages may contain non-text content, they still use SMS text data stream to communicate the data.

The great thing about Smart Messaging is that it allows people to do simple picture messaging, but with the low cost, quick message transmission and high simplicity of regular text messages. "Low cost", means that Smart Messages do not cost the mobile phone user any more money to send than a regular SMS text message, and they can be sent in roughly the same amount of time as SMS text messages.

"High simplicity", means that the user interface in the telephone makes creating Smart Messages very easy. IMU fully supports the Smart Messaging standard for text and picture messages. IMU discards any other content, such as ring tones and virtual business cards.

Concatenated SMS Messages / Long SMS Messages: One drawback of the SMS technology is that one SMS message can only carry a very limited amount of data. To overcome this drawback, an extension called concatenated SMS (also known as long SMS) was developed. A concatenated SMS text message can contain more than 160 English characters. Concatenated SMS works as: The sender's mobile phone breaks down a long message into smaller parts and sends each of them as a single SMS message.

When these SMS messages reach the destination, the recipient mobile phone will combine them back to one long message. The drawback of concatenated SMS is that it is less widely supported than SMS on wireless devices.

EMS (Enhanced Messaging Service): Besides the data size limitation, SMS has another major drawback - an SMS message cannot include rich-media content such as pictures, animations and melodies. EMS (Enhanced Messaging Service) was developed in response to this. It is an application-level extension of SMS. An EMS message can include pictures, animations and melodies. Also, the formatting of the text inside an EMS message is changeable. For example, the message sender can specify whether the text in an EMS message should be displayed in bold or italic, with a large font or a small font.

The drawback of EMS is that it is less widely supported than SMS on wireless devices. Also, many EMS-enabled wireless devices only support a subset of the features defined in the EMS specification. A certain EMS feature may be supported on one wireless device but not on the other.

There are many different kinds of SMS applications on the market today and many others are being developed. Applications in which SMS messaging can be utilized are virtually unlimited.

1.7.4. Applications of SMS

Person-to-Person Text Messaging: Person-to-person text messaging is the most commonly used SMS application and it is what the SMS technology was originally designed for. In these kinds of text messaging applications, a mobile user types an SMS text message using the keypad of his/her mobile phone, then he/she inputs the mobile phone number of the recipient and clicks a certain option on the screen, such as "Send" or "OK", to send the text message out. When the recipient mobile phone receives the SMS text message, it will notify the user by giving out a sound or vibrating. The user can read the SMS text message some time later or immediately and can send a text message back if he/she wants.

A chat application is another kind of person-to-person text messaging application that allows a group of people to exchange SMS text messages interactively. In a chat application, all SMS text messages sent and received are displayed on the mobile phone's screen in order of date and time. SMS text messages written by different mobile users may be displayed in different colors for better readability.

Provision of Information: A popular application of the SMS technology other than person-to-person text messaging is the provision of information to mobile users. Many content providers make use of SMS text messages to send information such as news, weather report and financial data to their subscribers. Many of these information services are not free.

Reverse billing SMS is a common way used by content providers to bill their users. The user is charged a certain fee for each reverse billing SMS message received. The fee will either be included in the monthly mobile phone bill or be deducted from prepaid card credits.

Downloading: SMS messages can carry binary data and so SMS can be used as the transport medium of wireless downloads. Objects such as ring tones, wallpapers, pictures and operator logos can be encoded in one or more SMS messages depending on the object's size. Like information services, wireless download services are usually not free and reverse billing SMS is a common way used by content providers to bill their customers.

The object to be downloaded is encoded in one or more reverse billing SMS messages. The mobile user who requests the object will be charged a certain fee for each reverse billing SMS message received.

If the mobile user is using a monthly mobile phone service plan, the download fee will be included in his/her next monthly bill; if the mobile user is using a prepaid SIM card, the download fee will be deducted from the prepaid credits.

Alerts and Notifications: SMS is a very suitable technology for delivering alerts and notifications of important events. This is because of two reasons:

A mobile phone is a device that is carried by its owner most of the time. Whenever an SMS text message is received, the mobile phone will notify us by giving out a sound or by vibrating. Users can check what the SMS text message contains immediately.

SMS technology allows the "push" of information. This is different from the "pull" model where a device has to poll the server regularly in order to check whether there is any new information. The "pull" model is less suitable for alert and notification applications, since it wastes bandwidth and increases server load.

1.7.5. SMS Alerts

Some common examples of SMS alert and notification applications are described below.

Email, Fax and Voice Message Notifications: In an email notification system, a server sends a text message to the user's mobile phone whenever an email arrives at the inbox. The SMS text message can include the sender's email address, the subject and the first few lines of the email body. An email notification system may allow the user to customize various filters so that an SMS alert is sent only if the email message contains certain keywords or if the email sender is an important person. The use cases for fax or voice message are similar.

E-commerce and Credit Card Transaction Alerts: Whenever an e-commerce or credit card transaction is made, the server sends a text message to the user's mobile phone. The user can know immediately whether any unauthorized transactions have been made.

Stock Market Alerts: In a stock market alert application, a program is constantly monitoring and analyzing the stock market. If a certain condition is satisfied, the program will send a text message to the user's mobile phone to notify him/her of the situation.

Remote System Monitoring: In a remote system monitoring application, a program (sometimes with the help of a group of sensors) is constantly monitoring the status of a remote system. If a certain condition is satisfied, the program will send a text message to the system administrator to notify him/her of the situation.

Two-way Interactive Text Messaging Applications: SMS messaging technology can be used as the underlying communication medium between wireless devices and servers in a two-way interactive text messaging application. For example, search engines are two-way interactive text messaging applications.

SMS Marketing: SMS messaging can be used as a marketing tool. An example is an SMS newsletter system. After signing up, the user will receive SMS text messages about the latest discounts and products of the company. If the user has any questions or comments, he/she can send a text message back with the questions or comments in it. The company may include its phone number in the SMS newsletter so that the user can talk to the customer service staff directly if he/she wants to do so.

An SMS center (SMSC) is responsible for handling the SMS operations of a wireless network. When an SMS message is sent from a mobile phone, it will reach an SMS center first.

The SMS center then forwards the SMS message towards the destination. An SMS message may need to pass through more than one network entity (e.g. SMSC and SMS gateway) before reaching the destination.

The main duty of an SMSC is to route SMS messages and regulate the process. If the recipient is unavailable (for example, when the mobile phone is switched off), the SMSC will store the SMS message. It will forward the SMS message when the recipient is available.

1.7.6. SMS Classification Based on its Operations

SMS is divided into so many types as and its types sent according to that working principle.

Intra-operator SMS Messages: If both we and our friend are using the mobile phone service of the same wireless network operator, the transmission of an SMS message from us to our friend will involve only one wireless network operator. This SMS message is called an intra-operator SMS message.

Typically, the cost for sending an intra-operator SMS message from a mobile phone is lower than that for sending other kinds of SMS messages such as inter-operator SMS messages. Some wireless network operators allow their subscribers to send unlimited intra-operator SMS messages free of charge.

The transmission of an intra-operator SMS message involves only one SMS center. After leaving the sender, the intra-operator SMS message reaches the SMS center. The SMS center then delivers the SMS message to the recipient mobile phone. If the recipient mobile phone is offline, the SMS center stores the SMS message. It will deliver the SMS message when the recipient mobile phone is online.

If the SMS message's validity period expires and the recipient mobile phone is still offline, the SMS center will remove the SMS message. When the SMS center receives the message delivery report from the recipient mobile phone or removes the SMS message (for example, when the validity period expires), it sends a status report to the sender if the sender requested one earlier.

Inter-operator SMS Messages

Suppose we and our friend are using the mobile phone service of wireless network operator A and wireless network operator B respectively. The transmission of an SMS message from us to our friend involves two wireless networks. This SMS message is called an inter-operator SMS message.

Typically, the cost for sending an inter-operator SMS message from a mobile phone is higher than that for sending an intra-operator SMS message. The transmission of an inter-operator SMS message involves one or more SMS centers. Generally, there are two different ways for the transmission of inter-operator SMS messages.

In the first way, signaling interconnections are set up between two wireless networks. When the originator SMS center receives an inter-operator SMS message, it gets the routing information from the recipient wireless network and delivers the SMS message to the recipient mobile phone directly.

1.7.7 Security Threats to SMS

Message Disclosure: Since there is no encryption applied by default in the short message during transmission, the messages may be eavesdropped by man-in-the-middle. In addition, the database for message storage may also be subject to attack.

Flooding/Spamming/DoS attack: Some SMS flooder programs are found to launch attack (e.g. flooding, DoS attack) by sending repeated messages to a mobile phone. The victim mobile phone will then become inaccessible. Both individuals and public SMS gateways may become victims. Examples of SMS flooder programs:

- TROJ_SMSMAX.20
- TROJ_FLOODER.A
- Hacktool.SMSDOS

SMS Phone Crash: Some vulnerable mobile phones may be crashed by receiving particular form of "problem" short message. Once the message is received, it is impossible to turn on an infected phone again.

SMS Virus: As phones are getting more intelligent and programmable, the potential to write virus becomes greater. Moreover, the ability of Subscriber Identity Module (SIM) application toolkit that allows applications to access to dialing functions and phone book entries makes SMS suits for writing self-replicating virus.

1.7.8. SMS Security Considerations

Message Transmission: When sending message via web browser, security protection should be in place during message transmission to prevent message disclosure. Take for instance, using SSL to secure the transmission. For those applications that require secure transmission of message, such as mobile banking, end-to-end encryption is advisable between the sender and the recipient. This could be done by means of SIM application toolkit. Launched in 1995, SIM application toolkit provides programming interface for applications running on SIM. It enables the SIM card to issue commands to the mobile phone and implement security critical applications through the use of encryption.

Storage Protection: For batch submission, message will be stored in database first before sending to the recipient. The database may not be a dedicated one that may be shared by different parties. Security of the database has to be considered. For storage of sensitive information, a central database server should be used.

Anti-SMS Bombing Filtering: Anti-SMS bombing filtering mechanism could be implemented in SMS gateway to block SMS flooding.

User Authentication: User login ID and password could be used to authenticate users to use the service for sending short messages. The user login ID and password should not be disclosed to others. But for secure transactions, the user authentication could be enhanced by means of digital certificates.

Protection of PC for Sending Message: For sending short message to SMS gateway via Internet, it is not advisable to use a public terminal. The PC used for sending the message should not be left unattended and should be equipped with protective measures such as anti-virus software.

1.7.9. Summary

Detailed discussion on SMS is given as: what is SMS, how it is transmitted as various types, the working principle of SMS and what are the threats to SMS then how these threats to be addressed.

CHAPTER 2

2. REVIEW OF LITERATURE

2.1. INTRODUCTION

This chapter explores all the existing or base works needed for our research work from various sources. In this, the existing data classification, transmission and what are the limitations in those transmissions are listed.

In the next part of this chapter, all the existing SMS, E-mail, Mobile instant messaging soft wares and their limitations are explained to spot out our work.

2.2. EXISTING DATA TRANSMISSION & CLASSIFICATIONS

Papers [51]-[54], are assumed two types of traffic in wireless networks: 1) Class I-real-time traffic, and 2) Class II—non real-time traffic. Class I traffic includes video and voice traffic from users equipped with an adjustable rate codec. In case of congestion, such users can gracefully adjust the coding rate such that the quality of video/audio.

In the papers [55]-[58], it is assumed that user movement patterns are known, and different amounts of bandwidth are reserved in different neighboring cells based on user movement patterns. A particular user's movement patterns can be provided. Various multimedia applications, six different application groups are assumed based on the connection duration, bandwidth requirement, and class of service (Class I or Class II).

Papers [59]-[63], assume the different application groups include constant bit rate (CBR), variable bit rate (VBR), and data traffic sources (unspecified bit rate—UBR).

In the scheme [64] two types of traffic with the admission control scheme, probabilities of receiving, rejecting calls and hand-off examined but Class II handoff connection results in a slower transmission rate and, thus results in longer transmission delay.

2.2.1. Limitations of Classified Data Transmission

- Non real time data connection results in a slower transmission rate
- The transmission delay for non real time data is longer.

2.3. EXISTING SMS SECURITY SOFTWARES

EmoSEC is a secure text messaging system [65] that is integral to the SIM card, designed and developed by the SIM card manufacturer, offering the user data the same level of integrity as that of the SIM operating system. With the addition of encryption, text messages are provided with unprecedented levels of security.

Whisper, a simple application that uses the Wireless Messaging API to send and receive encrypted SMS binary messages [66]. Two separate MIDlet suites were built using the Wireless Toolkit, one suite for Fleur and another for Viktor. Fleur's MIDlet suite contains Viktor's name, address and key file. The name and address are contained in the JAD file as the property: Whisper-Peer-1: Viktor|5550000

CryptoSMS is the only privacy system which uses three overlapping strong encryption schemes, employing both block and stream ciphers [67]. A "crypto" SMS is a secure Short Message that has been thrice encrypted using Blowfish over ARC4 over 3DEA, providing a triple layer Crypto Laminate composed of a stream cipher sandwiched between two block ciphers.

Unileadtone mingsoft Co.,ltd is a leader in the wireless and mobile and Computer Telephony that provides IVR/Voice Speech Message, SMS products and services that speeds up the delivery of next generation mobile/telephone applications to consumers and professionals alike:SMS, Email, telephone, Fax, CRM integration ,Barcode, SMS, telephone verify anti-count felting System ,Voice message ,SMS ,auto notify(birthday),bulk sending ,SMS logistics Tracking ,SMS machine Encrypting and OPT Man –decrypting [68] .

Easy Helper SMS Security helps us to hide and encrypt the secret SMS messages [69]. Hide secret SMS - With Easy Helper SMS Security, we can tune our phone with 4 different modes and choose among 7 different filters to hide and encrypt secret SMS messages.

CryptoGraf Messaging v2.0 S60 3rd Ed Send Message Keep Secret [70] is the best security software to encrypt mobile SMS/MMS to provide the Peer to Peer Privacy in Securing International Roaming.

Best Jotter allows jotting down notes, memos, any piece of information, organizing, categorizing, protecting and encrypting them with password if needed [71]. Users can have password to protect and encrypt our data stored in Best Jotter.

To enable security use 'Set Password' command. After the users specify the password all of their data will be encrypted. Every time users open Best Jotter, the application asks them for the password. When users work with the application all data encryption/decryption is performed on the fly and absolutely transparent to them.

Kryptext is a simple and intuitive program that enables encrypted SMS text messages to be sent and received in complete confidence [72]. With Kryptext every text message is encrypted during transfer and remains encrypted on our mobile phone. Nobody can decipher our private text messages.

SMS 007 system protects our data with strong algorithms SHA-2 and AES. CircleTech Corporation has developed brand new encryption software for our mobile phones –SMS 007 system [73, 74]. This system can protect our SMS messages against eavesdropping and other attacks. Even when our cellular phone has been stolen, SMS 007 will protect our received and sent messages against unauthorized reading.

XMS Mobile' is a software product for bringing confidentiality, integrity and non-repudiation to SMS messages [75]. It enables the user to send SMS to peers that are encrypted and optionally may be digitally signed.

2.4. EXISTING E-MAIL & CHAT SECURITY SOFTWARES

The paper [74] discusses about a 6144 BIT ENCRYPTION SOFTWARE-SHYFILE with the following characteristics:

- Make up a 32 character key entry
- Enter the text we wish to encode
- Attach secure ShyFile to our email
- Recipient simply uses a browser to decode

Chat Encrypter Pro helps us to encrypt messenger's messages [76]. We can choose the messenger (AIM, ICQ, MSN, and Yahoo messenger), the encryption algorithm (3DES, Blowfish, Skipjack) and the password. Then choose user and communicate with he/she privately. Main Chat Encrypter Pro features: AIM messenger support; ICQ messenger support; MSN messenger support; Yahoo messenger support; 3DES algorithm support; Blowfish algorithm support; Skipjack algorithm support; User friendly interface.

2.5. LIMITATIONS OF EXISTING SMS & E-MAIL SECURITY SOFTWARES

- Phone Number/Fixed Constant is used as Key to encrypt or decrypt.
- The Encryption time is greater than 2secs.
- Application size is large and it should be stored in phone memory which reduces the space in phone memory.
- Keys are given or passed as it is without any modification which in turn gives the possibility of hacking it.
- If the low end device receives SMS from high end device then the synchronization of message division is needed. i.e 450 characters of one SMS sent as 160 characters of three message

2.6. SUMMARY

This chapter explains all the related and base open works for our research and pointed out their limitations which are to overcome.

CHAPTER 3

3.1. INTRODUCTION

Security in wireless networks is a complex thing. Security in wireless networks is an important issue since users are likely to put personal, important or mission-critical data over an infrastructure that is not truly secure [77]. The security weaknesses stem from both using multiple incompatible security schemes and design flaws in security protocols, which is inherent. The greatest danger is that the user may perceive the entire structure as secure and may mistakenly trust it to convey confidential information.

Therefore, the security strategy must be devised and implemented with respect to the type of data being transported and the estimated loss in case of eavesdropping or tampering with the data. We have to also consider the fact that many security issues arise due to poor implementation, feature interactions, unplanned growth and new flaws created due to prior attacks. To be truly effective, the security strategy must be applied end-to-end, i.e. from source to destination regardless of path.

With respect to security, we have emphasized the obscurity that surrounds the protocols used for authentication and encryption in GSM networks. This scheme overcomes the above said limitations.

3.2. SCHEME

In 2004, Wang etal.'s attacks on MD4, MD5, HAVAL, and RIPEMD [92, 93] and SHA-0/1 [94, 95] brought the big impact on the field of symmetric key cryptography including hash function. RIPEMD-family [86] has somewhat different approach for designing a secure hash function.

We have proposed an end-to-end cryptography algorithm for non real time data like SMS, E-mail, instant messaging. At the first phase, we have devised an algorithm for GSM – 2G mobiles which do not have data packetization. In this scheme, the data whatever to send or receive to or from the computer and mobile is divided into packets in source it self and then encrypted or decrypted at the respective places.

In the second phase, we have considered 3G mobiles, in which the non real time data whatever is sent or received, are covered with our private algorithm in the source itself.

3.2.1. Objectives

The objectives of this research work are protecting the private SMS in the following situations:

(i) Since the SMS content is delivered in plain text and binary fields, the crackers can easily intercept the communication .An intruder can set up his own fake gateway which can then send all kinds of malicious short messages to the MS users through the SMSC.

(ii) Eavesdropping and Modifying are possible to implement just like in the case of normal GSM phone calls.

(iii) Synchronize the size of the messages.

(iv) Reducing the memory size and the delay.

(v) To improve the transmission rate

3.2.2. Working Principle of algorithm for GSM - 2G MOBILES

In our algorithm, a private key is framed at the servers and compared with message to swap the data by performing XOR and compression functions with the DNA set [100]. This algorithm will create a text file and store the packets what to send or receive into it. The following line is to be executed at each server or sub servers to precede the transmission of data with this private security algorithm.

```
txtfile=gprs(gcsec(message));
```

The data can be secured by the security algorithm as one character of the input is replaced by four characters which are stored in a text file as blocks of size 72 * 24(1728 bytes). Each block has its own header in the following format:

Input file name with its extension (12 characters - if not pad # character)	Input file size in hexadecimal representation(4 characters)	Current block number (4 characters - if not pad # character)	Total number of blocks (8 characters)

Each line of the compressed block of the text file is considered as a packet and sent with the above header to the next which establishes the packet switching network to reduce the connection and transmission time of a network which in turn reduces the transmission delay.

GCSEC ALGORITHM:

```
GCSEC(data){
 Each character of the input is XORed with each character of key;
If the length (key) < length (input), then repeat the following for all characters:
Do the right rotation followed by one transposition of the key;
XORed results of each character passed to encode() function;
Output always stored in text file as blocks which in turn compressed;
}
```

```
encode()
{
Convert the resultant character into 8 bit representation;
Select two MSB of 8 bits and pass as index to A/C/G/T set and result will be the first character of the encoded string;
Remaining 6 bits will be converted as index to DNA character set;
Return all 4 characters as output;     }
```

Algorithm for Packet Division

```
gprss()
{
Each line of the compressed file considered as packet;
padd with the header };
```

This algorithm is called as a C++ networking function in network simulator 2.2.6 under Linux server edition for simulation. This simulation yields packetization of data but it has taken some more time to install and execute this procedure at server. Then this software is to be active at all times in all servers or sub servers of each mobile switching station or centre for data transmission. Hence the transaction cost is also increased by the above operations.

3.2.3. Working Principle of algorithm for 3G MOBILES

To overcome the limitations of first phase, we have taken non real time data security in the source or at the mobile station itself for 3G mobiles. Due to the advancement in the field of data communication through computer and mobile networks like 3G, problems arises with the security of data.

In the context of communications across a network, the following attacks can be identified [98]: Disclosure, Traffic analysis, Masquerade, Content modification, Sequence modification, Timing modification, Repudiation. These attack to be removed using this scheme.

Authentication is the remedy to the other attacks. User Authentication is defined as 'Provision of Assurance that the message is originated from authorized user'. Message Authentication is defined as 'Provision of assurance that the message is not altered' [99]. One type of Message Authentication is by hash algorithm.

This provides an assurance to the destination that the message is not changed by the intruders. To get protected from eavesdropping, encryption in source and decryption in destination is also done with the help of a secret key. Encryption can be used to ensure secrecy, but other techniques are still needed to make communications secure, particularly to verify the integrity and authenticity of a message.

Providing a way to check the integrity of information transmitted over or stored in an unreliable medium is a prime necessity in the world of open computing and communications. Mechanisms that provide such integrity checks based on a secret key are usually called message authentication codes.

Typically, message authentication codes are used between two parties that share a secret key in order to authenticate information transmitted between these parties. This standard defines a message authentication code that uses a cryptographic hash function in conjunction with a secret key.

In order to overcome the above said limitations of existing systems, this scheme provides an end to end private algorithm in two dependent layers. Output of the first layer is given as input to the second layer as in figure 4. The first layer hashes the private password stored in the software as argument of the encrypted software stored in phone or card memory.

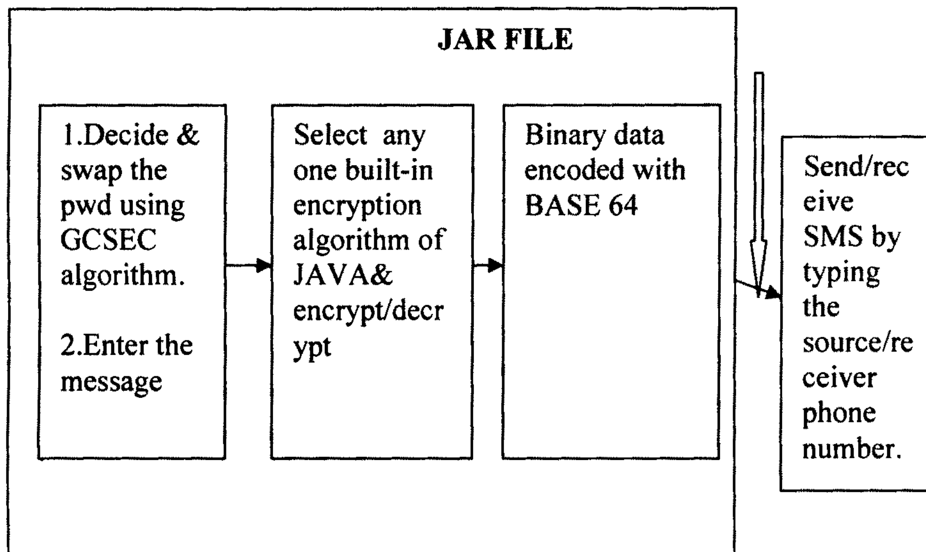


Figure: 4 Block Diagram of GCSEC algorithm

GCSEC Algorithm

Let $\oplus$ represent Exclusive OR, $\boxplus$ represent addition modulo 232. A, B, C, D, E, F, G and H are the initial chaining variables. These variables are modified over time with new blocks of data being processed, and the resultant chaining variables form the hash. Most dedicated hash functions which have iterative process use the Merkle-Damgard construction [83, 87] in order to hash inputs of arbitrary length.

A key in communication is processed by 512-bit block. GCSEC hashes a 512-bit string to a 256-bit string. It consists of five parallel branch functions, BRANCH1, BRANCH2, BRANCH3, BRANCH4 and BRANCH5.

Let CVi = (A, B, C, D, E, F, G, and H) be the chaining variable of the density function [97]. It is initialized to IV0 which is:

A = 6a09e667x   B = bb67ae85x

C = 3c6ef372x   D = a54ff53ax

E = 510e527fx   F = 9b05688cx

G = 1f83d9abx   H = 5be0cd19x.

Each consecutive 512-bit message block M is divided into sixteen 32-bit words M0, M1, · · ·, M15 and the following computation are performed to update CVi to CVi+1 which is given in the algorithm gush.

$Z=[BRANCH1(CVi, \sum_1(M)) + BRANCH2(CVi, \sum_2(M))]$

$Y= [BRANCH2 (CVi, \sum_2(M)) + BRANCH3(CVi, \sum 3(M))]$

$X= [BRANCH3 (CVi, \sum_3(M)) + BRANCH4(CVi, \sum_4(M))]$

$X1=[BRANCH4(CVi,\sum_4(M)) + BRANCH5(CVi, \sum 5(M))]$

$X2= Z+ Y$

$X3= X+X1$

$CVi+1 = CVi + [X2 \oplus X3]$

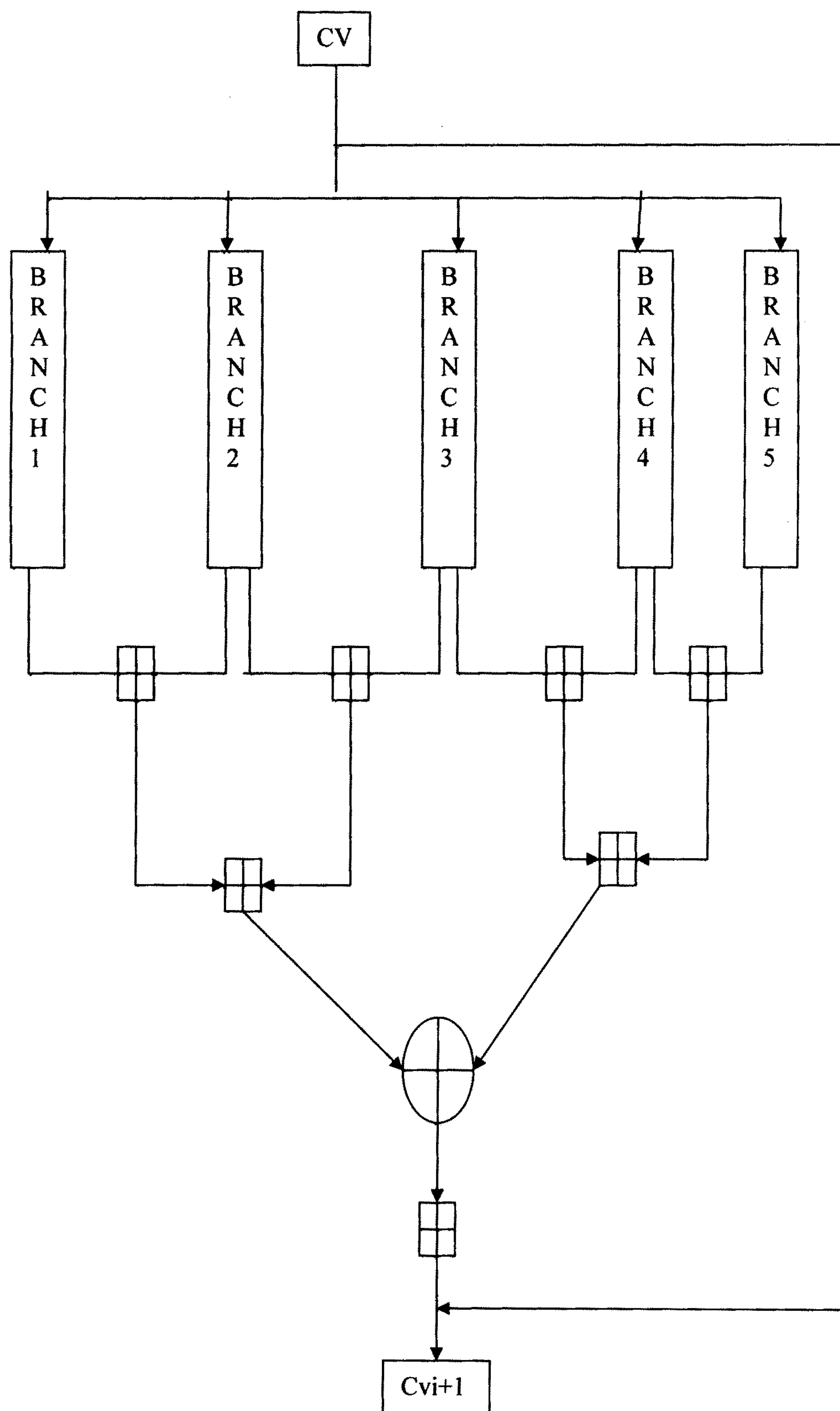
where, $\sum i(M)$ gives the random selection of any part the password of 512 bits either padded with zeroes at end if its length is less than 512

bits or the password as it is . This password is limited to 64 any alphanumeric characters or the symbols allowed in mobiles. Sixteen constants are initialized to

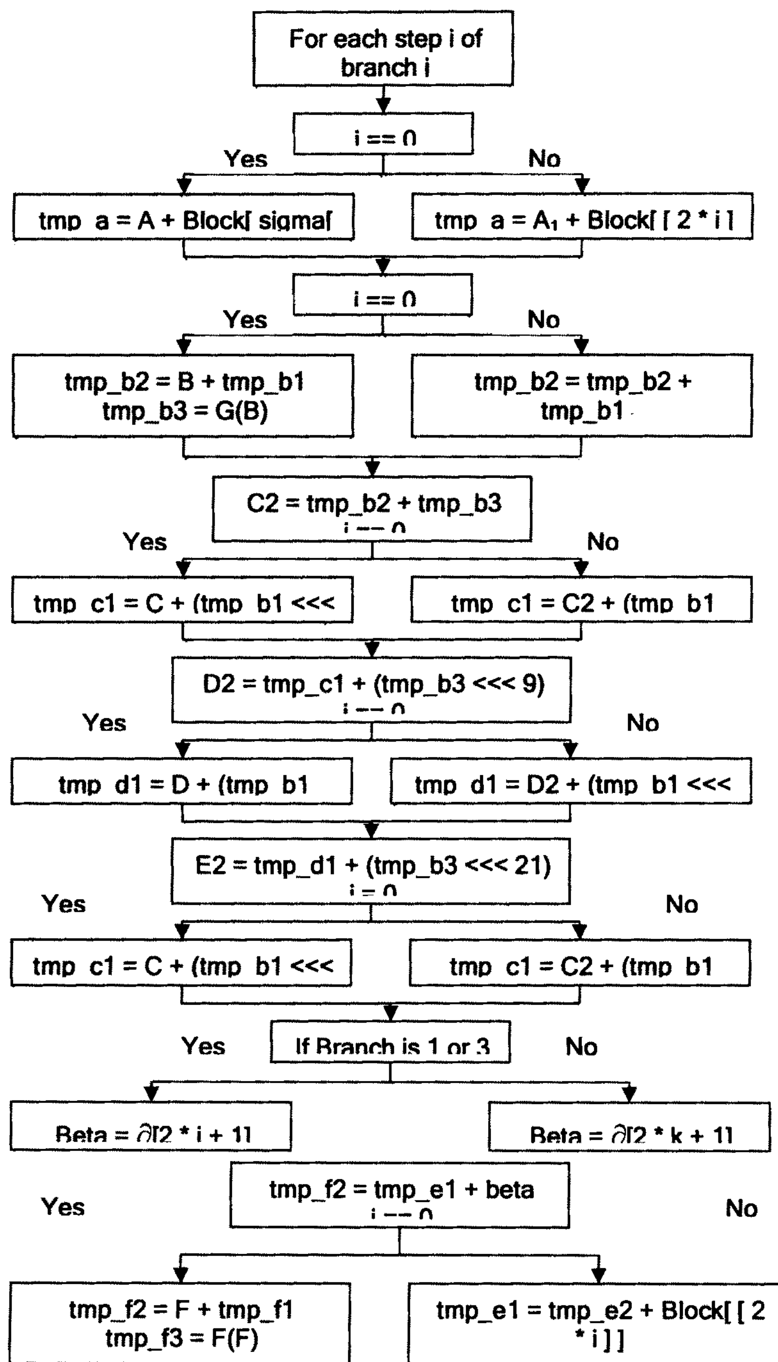
C_0 = $428a2f98_x$	C_1 = 71374491_x
C_2 = $b5c0fbcf_x$	C_3 = $e9b5dba5_x$
C_4 = $3956c25b_x$	C_5= $59f111f1_x$
C_6 = $923f82a4_x$	C_7= $ab1c5ed5_x$
C_8 = $d807aa98_x$	C_9= $12835b01_x$
C_{10} = $243185be_x$	C_{11}= $550c7dc3_x$
C_{12} = $72be5d74_x$	C_{13}= $80deb1fe_x$
C_{14} = $9bdc06a7_x$	C_{15}= $c19bf174_x$

By using constants users pursue the goal to disturb the attacker who tries to find a good differential characteristic with a relatively high probability [97]. At each branch, any one of these constants randomly selected and added with the random part of the password and the result is shifted left and then stored into the chaining variables for next branch computations. The exact computation is elaborated in the branch computations.

The preimage resistance of this algorithm is 2^{256} and the collision resistance is 2^{-256}. The security level of this algorithm is increased by the parallel branch operations, complex XORs and shift operations. There is no possibility of guessing the constant values by any intruders which gives us more security considerations. Its structure should be resistant against known attacks including Wang etal.'s attack [78–82, 84, 85, 91–95, 96].



Algorithm gush



Branch Computation

3.2.4. Etiquette Steps -First Phase

The algorithm steps of the first layer are given as:

1. Enter the password.
2. Check the length of password .If it is <512 bits, pad 0's to make that password of 512 bits.
3. Else, divide this password into 16 words (w1...w15) of 32 bits.
4. Store 16 constants with the some hexadecimal values(c1...c15)
5. Initialize 8 chaining variables (A...H) to store intermediate results.
6. Randomly select both the constant & the word to be swapped after performing a sequence of XOR and left shift (5/7/9/21/17/19times depends on iterations) operations and this result is stored in the chaining variables.
7. The current result is added with the previous value to get the updated current value.
8. Repeat the steps 6&7 for all words until the password is hashed into 256 bits.

This part first encrypts the message with the hashed password and the encrypted data in turn encoded using BASE 64 in order to avoid sending SMS or the instant messages or e-mails in binary format. Since the binary format of non real time data allows the intruders to hack it or they can introduce the fake gateway to filter the passing data from their switching centers itself.

The second layer receives the hashed password of 256bits as key to encrypt or decrypt the message using built in data encryption algorithms of J2SE at the respective places dynamically.

3.2.5. Etiquette Steps- Second Phase

1. message = message to send/receive + hashed password
2. Message passed to any one of the cryptographic algorithms by selecting java.security.algorithmname, which results in encrypted or decrypted message in the respective places.
3. The resultant data is in binary, hence it is in turn encoded using BASE 64 and connected as SMS using J2ME.

3.2.6 Advantages

- Private password stored inside the software itself.
- Password can't even be guessed by anybody since it is hashed
- Data reduced to 50% of it.
- Data Encryption algorithm is selected dynamically.
- SMS/E-mail/Instant message can be encrypted or decrypted respectively using an end to end cryptography.

3.3. CONTRIBUTIONS OF THIS RESEARCH WORK

1. Our algorithm provides the private cover for sending or receiving non real time data like e-mail, SMS/instant messages in mobiles of S60 series, NOKIA 7710 & N series and various system processors with less transmission delay.

2. This security algorithm is more complex because of many numbers of steps for hashing the left and right parts of the one portion of data for reducing hack able factors.

3. GCSEC is having the highest level of privacy because of the number of complex XOR and shift operations which in turn introduces the less probability even for assuming the password.

4. Our algorithm provides client end-to-end privacy for all types of data.

5. Our algorithm is the secured platform for S60 mobiles to send/receive any data.

CHAPTER 4

4. ANALYSIS

In the software field, any problem can get a better and efficient solution by performing a schematic analysis. This phase yields an immediate solution.

4.1. INTRODUCTION

Some system vulnerabilities arise due to errors in individual system components: e.g., buffer overflow attacks are aimed at memory errors in server processes. However, a majority of vulnerabilities arise due to interactions among several components such as the operating system kernel; file system, server processes, etc. For instance, consider a vulnerability that existed in early versions of the fingered service. In servicing a query "finger username," this program needs to read a file named .plan in the home directory of the
user username.

A malicious user u could symbolically link a file f as his/her .plan even if the user has no read access to f. Consider the vulnerability involving the mail notification program comsat, which waits for reports of incoming mail for any user and prints the first few lines of the message on the terminal in which the user is logged on.

This terminal is determined from the file /etc/utmp, which was configured to be world-writable. A malicious user could modify this file by substituting the /etc/passwd in the place of the terminal that he/she is logged on. The user then sends mail to self containing a line that starts with root: 0:0:, the second field in this line corresponds to the password field, is empty, which implies that the super user has no password. The comsat program promptly overwrites the password file with the message.

The user can now login as root without providing a password. This second vulnerability also arose from an interaction among several components: the comsat program that assumed the correctness of /etc/utmp file, the file system, and the mail delivery program.

More generally, many vulnerabilities arise from unexpected interactions between different components, violation of hidden assumptions, improper setting of system parameters and configurations, etc.

Use of good software engineering practices has the potential to eliminate some of these vulnerabilities, but given the fact that new vulnerabilities continue to surface in UNIX server programs that have been operational for well over a decade, that clearly need alternative mechanisms to guard against vulnerabilities. Consequently, several recent research efforts have focused on vulnerability analysis and intrusion detection techniques.

4.2. VULNERABILITY ANALYSIS

Efforts in software security analysis fall into: Vulnerability analysis, static code analysis, security testing, formal verification, and security evaluation standard methodologies. Vulnerability analysis mainly refers to efforts directed towards classification of security bugs. Static code analysis can be used to find security-related errors.

Several methods exist that could be manual as in code inspection or automated using tools. In security testing, techniques of property-based testing are mostly used. The formal verification methods can be used for the verification of security properties. Our system performs number of checks to provide certain level of assurance.

4.2.1. Networking Vulnerabilities

MIDP SSL Vulnerability: In order to establish a secure connection with remote sites (HTTPS), the reference implementation of MIDP uses SSL v3.0 protocol. During the SSL handshake, the protocol has to generate random values to be used to compute the master secret.

The master secret is then used to generate the set of symmetric keys for encryption. Hence, generating random values that are unpredictable is an important security aspect of SSL. It is well known that the challenge in producing good random values is how to update the seed. The seed is an initial value on which we apply a certain algorithm in order to generate random values.

Generating a set of random values occurs in the following way: the current seed value is used to generate a random value, then, the seed is updated and a second random value can be generated and so on. By inspecting the implementation of our system, we have noticed that the seed update depends only on the system time.

Hence, in order to obtain the random value generated by the client, all what the attacker has to do is to guess the precise system time (in milliseconds) at the moment of the random value computation [101].

Unauthorized SMS Sending Vulnerability: As every security-sensitive API, Wireless Message API (WMA), allowing the exchange of SMS messages, requires appropriate permissions to be used. Usually, user permission is obtained through an on-screen dialog. That is, when a program needs to send an SMS message, the device displays a dialog asking the user whether he accepts to send the SMS message and hence to assume charges.

Consequently, sending an SMS message without the authorization of the user is considered as a security flaw. As mentioned earlier, the Penult hackers group has discovered that the Siemens S55 phone has a vulnerability that makes the device send SMS messages without the authorization of the user. Hence SMS authorization dialog is executed before sending SMS in our system.

4.2.2. Storage System Vulnerabilities

Managing the Available Free Persistent Storage Vulnerability: Embedded devices have limited memory resources hence no restrictions are made on the persistent storage granted to one MIDlet, we can not prevent any MIDlet from getting all the available free space on the persistent storage for its record stores.

Unprotected Internal APIs Vulnerability: MIDP APIs were designed in several levels of abstraction. The highest level contains all what a developer needs in order to develop MIDlets. The low level APIs are closer to the device hardware, and therefore more difficult to program, but they have more privileges and less restrictions.

Retrieving and Transferring JAR Files from a Device: MIDlets are transferred from one device to another in our system as JAR file since in Series 60 phones JAD and JAR files are typically stored in the typical directory.

Retrieving and Transferring MIDlet Persistent Data: In addition to JAR and JAD files, using FExplorer software, it is possible to transfer MIDlet persistent data from a device to another.

4.2.3. Threading System Vulnerabilities

J2ME CLDC supports multithreading and these threading systems were analyzed and vulnerabilities were discovered.

4.3. DESIGN ANALYSIS

Comparison of designs

The various hash algorithms are analyzed with our algorithm's design and it is explained in the following table.

	MD5	SHA	RIPEMD	GCSEC
Digest length	128 bits	160 or 256 or 512	160 bits bits	256 BITS
Basic unit of processing	512	512	512	512
Number of steps	64	80	160	1312
Maximum message size	∞	$2^{64}-1$ bits	$2^{64}-1$ bits	∞
Primitive logical function	4	4	5	Nil
Additive constants used	0	4	9	16
Endian format	Little-endian	Big-endian	Little-endian	Little-endian

4.4. TOOLS ANALYSIS

This chapter elaborately explains all the tools and their features used to simulate our algorithm.

4.4.1. J2ME

J2ME stands for Java 2, Micro Edition. It is a stripped-down version of Java targeted at devices which have limited processing power and storage capabilities and intermittent or fairly low-bandwidth network connections [102]. These include mobile phones, pagers, wireless devices and set-top boxes among others. J2ME combines a resource constrained JVM and a set of Java APIs for developing applications for mobile devices.

Sun Microsystems has defined three Java platforms, each of which addresses the needs of different computing environments:

- Java 2 Standard Edition (J2SE)
- Java 2 Enterprise Edition (J2EE)
- Java 2 Micro Edition (J2ME)

The inception of the J2ME platform arose from the need to define a computing platform that could accommodate consumer electronics and embedded devices. These devices are sometimes referred to collectively as pervasive devices. The creators of the J2ME platform delineated pervasive devices into two distinct categories:

- **Personal, mobile information devices** that are capable of intermittent networked communications—mobile phones, two-way pagers, personal digital assistants (PDAs), and organizers

- **Shared-connection information devices** connected by fixed, uninterrupted network connection—set-top boxes, Internet TVs, Internet-enabled screen phones, high-end communicators, and car entertainment /navigation systems.

The first category describes devices that have a special purpose or are limited in function; they are not general-purpose computing machines. The second category describes devices that generally have greater capability for user interface (UI) facilities.

Of course, devices with superior UI facilities typically have more computing power. Practically speaking, computing power is the primary attribute that distinguishes these two categories of devices. Nevertheless, this delineation is somewhat fuzzy, because technology continues to enable more and more power to be placed in smaller and smaller devices.

Like computing power, connectivity—the availability of media such as wireless networks—also affects the kinds of functionality and services that pervasive devices can support. The challenge and the primary goal for J2ME are to specify a platform that can support a reasonable set of services for a broad spectrum of devices that have a wide range of different capabilities.

The creators of J2ME identify modular design as the key mechanism that enables support for multiple types of devices. The J2ME designers use configurations and profiles to make J2ME modular.

Defining a Java Platform for Pervasive Devices: Configurations and profiles are the main elements that comprise J2ME's modular design. These two elements enable support for the plethora of devices that J2ME supports.

A J2ME configuration defines a minimum Java platform for a family of devices. Members of a given family all have similar requirements for memory and processing power. A configuration is really a specification that identifies the system-level facilities available, such as a set of Java language features, the characteristics and features of the virtual machine present, and the minimum Java libraries that are supported [103]. Software developers can expect a certain level of system support to be available for a family of devices that uses a particular configuration.

A configuration also specifies a minimum set of features for a category of devices. Device manufacturers implement profiles to provide a real platform for a family of devices that have the capabilities that a given configuration specifies.

The other J2ME building block, the profile, specifies the application-level interface for a particular class of devices. A profile implementation consists of a set of Java class libraries that provide this application-level interface. Thus, a profile theoretically could specify all kinds of functionality and services.

For example, a profile might support a network communication facility for the popular Short Message Service (SMS) standard widely used by mobile phones. Because the SMS standard is a ubiquitous feature of mobile telephony, it makes sense to define this service in a profile that targets mobile phones, rather than to build it into a configuration.

A profile is implemented on top of a configuration, one step closer to the implementation of real-world applications. Typically, a profile includes libraries that are more specific to the characteristics of the category of devices they represent than are the libraries that comprise configurations. Applications are then built on top of the configuration and profile; they can use only the class libraries provided by these two lower-level specifications. Profiles can be built on top of one another.

A J2ME platform implementation, however, can contain only one configuration. Figure.6 shows the conceptual layers that comprise the J2ME platform

Configurations and Profiles: A configuration specifies three basic elements:

- a set of Java programming language features
- a set of Java virtual machine features
- a set of supported Java libraries and application programming interfaces (APIs)

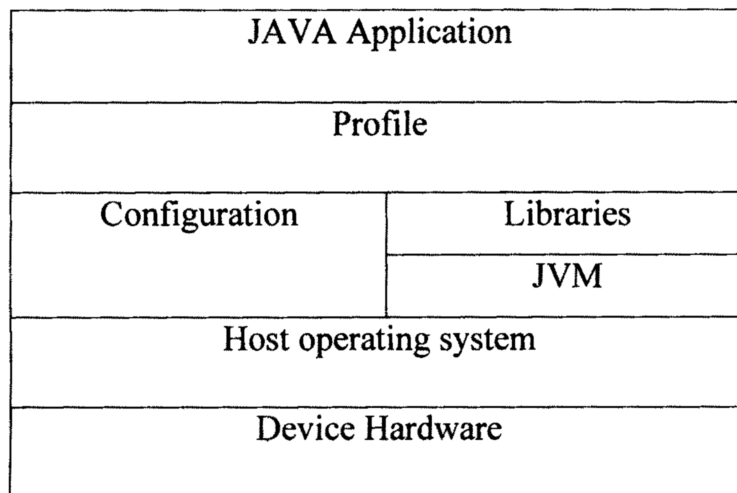


Figure.5 J2ME platform

The J2ME platform consists of a set of layers that support a basic runtime environment with core Java libraries and a Virtual Machine (VM), a set of system-level application programming interfaces (APIs) in a configuration, and a set of application-level APIs in a profile.

The creators of J2ME have defined only two configurations to avoid a fragmented landscape of incompatible platforms. The two configurations that exist currently represent the two categories of pervasive devices we saw earlier in this chapter, namely:

- **personal, intermittently connected mobile devices**—supported by the Connected, Limited Device Configuration (CLDC)
- **constantly connected network devices**—supported by the Connected Device Configuration (CDC)

Theoretically, a configuration could specify the very same support as the J2SE platform libraries. This is unlikely in the real world because, as we now know, J2ME is targeted at devices that are far less powerful than desktop computers.

Configuration specifications require that all Java classes adapted from J2SE be the same as or a proper subset of the original J2SE class. That is, a class cannot add methods not found in the J2SE version. Configurations can include additional classes in their specifications, however; configurations themselves are not necessarily proper subsets of J2SE. Both configurations that have been defined to date add classes not present in J2SE in order to address device attributes and constraints.

The Connected Device Configuration (CDC): The Connected Device Configuration (CDC) intends to capture just the essential capabilities of each kind of device in the category of devices it targets, namely, devices with 2 MB or more of total memory, including both RAM and ROM.

As in Figure 5, a configuration specifies both the set of Java VM features that are supported and a set of class libraries. The CDC specifies the use of the full Java 2 platform VM, which, in this context, is called the Compact Virtual Machine (CVM).

The CVM: Although the CVM supports the same features as the J2SE VM, it is designed for consumer and embedded devices. This means that the standard J2SE VM has been reengineered to suit the constraints of limited-resource devices. The features of the resulting offspring CVM are:

- advanced memory system
- small average garbage collection pause times
- full separation of VM from memory system
- modularized garbage collectors
- generational garbage collection

In particular, the CVM has been engineered to offer the following features:

- portability
- fast synchronization
- execution of Java classes out of read-only memory (ROM)
- native thread support
- small class footprint
- provision of interfaces to and support for real-time operating system (RTOS) services
- mapping Java threads directly to native threads
- support for all Java 2, v1.3 VM features and libraries: security, weak references, Java Native Interface (JNI), Remote Method Invocation (RMI), Java Virtual Machine Debugging Interface (JVMDI)

The CDC specifies a minimal set of class libraries and APIs. It supports the following standard Java packages:

java.lang—Java VM system classes
java.util—underlying Java utilities
java.net—Universal Datagram Protocol (UDP) datagram
and input/output (I/O)
java.io—Java files I/O
java.text—very minimal support for internationalization
java.security—minimal fine-grain security and encryption
for object serialization.

These APIs do not include the full set of Java 2 software development kit (SDK) packages. In some cases, these packages and classes are subsets of the Java 2 SDK packages and classes. Table .1 lists the full set of packages supported by the CDC.

Table .1 CDC Packages

CDC Package Name	Description
java.io	Standard IO classes and interfaces
java.lang	VM classes
java.lang.ref	Reference classes
java.lang.reflect	Reflection classes and interfaces
java.math	Math package
java.net	Networking classes and interfaces
java.security	Security classes and interfaces
java.security.cert	Security certificate classes
java.text	Text package
java.util	Standard utility classes
java.util.jar	Java Archive (JAR) utility classes
java.util.zip	ZIP utility classes
javax.microedition.io	CDC generic connection framework classes and interfaces

The Foundation Profile: A configuration, together with a profile, creates a J2ME runtime environment. The system-level features and services supported by a configuration are more or less hidden from the application developer. In reality, the application developer is prohibited from accessing them directly. If this were not the case, the application would not be considered J2ME compliant.

From the programmer's perspective, a profile is required to do "useful" work. A profile defines the layer that contains the APIs that the programmer usually manipulates. The J2ME creators initially defined one CDC profile, the Foundation Profile, which is based on the J2SE v1.3 release. It was designed by standard committee through the Java Community Process, by an expert group of companies in the consumer electronics industry. The Foundation Profile contains the J2SE packages listed in Table.2.

Notice that the whole java.awt Abstract Window Toolkit (AWT) and javax.swing Swing package hierarchies that define the J2SE graphical user interface (GUI) APIs are absent from the supported packages. If an application needs a GUI, an additional profile would be required. Profiles can be built on top of one another. An implementation of the J2ME platform, however, can contain only one configuration.

The lack of GUI support in the Foundation Profile has less impact for the family of shared, constantly connected network devices such as TV set-top boxes than it does for personal, mobile devices, which are served by the second J2ME configuration, the CLDC.

In general, the decision to include or omit features and libraries from a configuration or profile is based on their footprints, static and dynamic resource requirements, and security requirements.

Table.2 Foundation Profile Packages - CLDC

Foundation Profile Package Name	**Description**
java.lang	Rounds out full **java.lang.*** J2SE package support for the Java language (**Compiler**, **UnknownError**)
java.util	Adds full zip support and other J2SE utilities (**java.util.Timer**)
java.net	Adds TCP/IP Socket and HTTP connections
java.io	Rounds out full **java.io.*** J2SE package support for Java language input/output (**Reader** and **Writer** classes)
java.text	Rounds out full **java.text.*** J2SE package support for internationalization (I18N): **Annotation**, **Collator**, **Iterator**
java.security	Adds code signing and certificates

The Personal Profile specification was created through the Java Community Process, resulting in JSR–62. The Personal Profile provides an environment with full AWT support. The intention of its creators is to provide a platform suitable for Web applets. It also provides a J2ME migration path for Personal Java applications.

Personal Profile version 1.0 requires an implementation of the Foundation Profile version 1.0. It is a superset of the Personal Basis Profile version 1.0. Personal Profile is a subset of the J2SE version 1.3.1 platform, however, which makes Personal Profile applications upward compatible with J2SE version 1.3.1.Table 1.3 lists the packages that comprise Personal Profile version 1.0.

Table.3 Foundation Profile Packages- CDC	
Personal Profile Package Name	**Description**
java.applet	Classes needed to create applets and those used by applets
java.awt	Classes for creating AWT UI programs
java.awt.datatransfer	Classes and interfaces for transferring data within and between applications
java.awt.event	Classes and interfaces for AWT event handling
java.awt.font	Classes and interface for font manipulation
java.awt.im	Classes and interfaces for defining input method editors
java.awt.im.spi	Interfaces that aid in the development of input method editors for any Java runtime environment
java.awt.image	Classes for creating and modifying images
java.beans	Classes that support JavaBean development
javax.microedition.xlet	Interfaces used by J2ME Personal Profile applications and application managers for communication

The RMI Profile is a profile designed for platforms that support the CDC configuration. It has been defined by JSR-66 by various companies participating through the Java Community Process.

The RMI Profile requires an implementation of the Foundation Profile and is built on top of it. RMI Profile implementations must support the following features:

- full RMI call semantics
- marshaled object support
- RMI wire protocol
- export of remote objects through the UnicastRemoteObject API
- distributed garbage collection and garbage collector interfaces for both client and server side
- the activator interface and the client side activation protocol
- RMI registry interfaces and export of a registry remote object

The RMI profile supports a subset of the J2SE v1.3 RMI API. The following interfaces and features are part of the J2SE v1.3 RMI specification and public API, but support for these interfaces and functionality is omitted from the RMI profile specification because of limitations on device processing power, network performance, and throughput:

- RMI through firewalls and proxies
- RMI multiplexing protocol
- implementation model for an "activatable" remote object
- deprecated methods, classes, and interfaces
- support for the RMI v1.1 skeleton/stub protocol
- stub and skeleton compiler

Support for the following J2SE RMI v1.3 properties is omitted:

java.rmi.server.disableHttp
java.rmi.activation.port
java.rmi.loader.packagePrefix
java.rmi.registry.packagePrefix
java.rmi.server.packagePrefix

Connected, Limited Device Configuration (CLDC): The second of the two J2ME configurations, the Connected, Limited Device Configuration (CLDC), supports personal, mobile devices, which constitute a significantly less powerful class of devices than the one that the CDC supports [105]. The CLDC specification identifies devices in this category as having the following characteristics:

- 160 to 512 KB total memory available for the Java platform
- 16-bit or 32-bit processor
- low power consumption, often battery powered
- intermittent network connectivity (often wireless) with potentially limited bandwidth

The goal of the CLDC is to define a standard Java platform for these devices. Because of the wide variety of system software on various personal devices, the CLDC makes minimum assumptions about the environment in which it exists. For example, one OS might support multiple concurrent processes; another might or might not support a file system, and so forth.

The CLDC is different from, yet also a subset of the CDC. The two configurations are independent of each other, however, so they should not be used together to define a platform. Figure 6 shows the relationship between the two configurations and the J2SE platform.

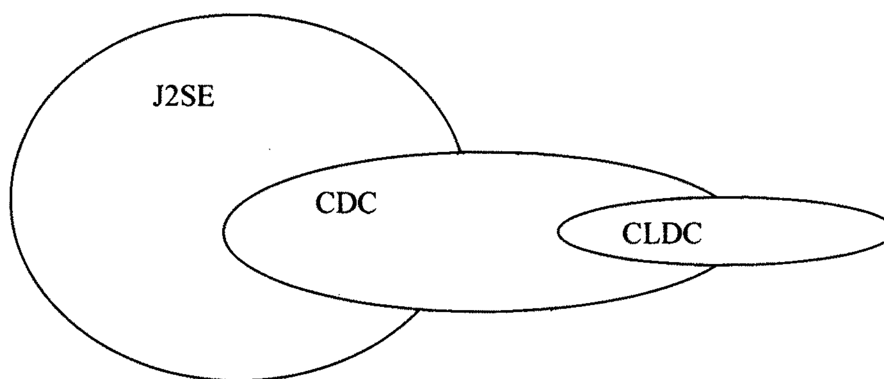


Figure.6 J2SE and CLDC relationship

The CLDC is a proper subset of the CDC. Neither the CLDC nor the CDC is a proper subset of the J2SE platform, however, because both of these configurations add new classes necessary to deliver services on their respective families of devices.

Like the CDC, the CLDC specifies the level of support of the Java programming language required, the required functional support of a compliant Java VM, and the set of class libraries required.

Java Language Support: The CLDC specification omits support for the following features of the Java language:

- floating point calculations
- object finalization
- The java.lang. Error class hierarchy in its entirety

The lack of floating point support is the main language-level difference between a Java virtual machine that supports CLDC and a standard J2SE VM that is visible to programmers. This means that programs intended to run on the CLDC cannot use floating point literals, types, or values. The float built-in type can't be used, and the java.lang. Float class has been removed from CLDC libraries. This feature is not present because of the lack of floating-point hardware or software on most mobile devices.

Object finalization is also absent. This means that the Object.finalize() method has been removed from the CLDC libraries. The java.lang.Error exception hierarchy has also been removed from the CLDC libraries and is therefore not available to applications.

And the resource cost of implementing error handling is expensive, beyond the capabilities of today's mobile devices. Moreover, error recovery is device-specific on embedded devices like mobile phones. In consequence, it doesn't make sense to stipulate the recovery mechanism that devices should use. This mechanism may well be outside the scope of an embedded VM.

Java Virtual Machine and Library Support: The CLDC specifies requirements for a Java virtual machine. It defines a VM that is highly portable and designed for resource-constrained small devices. Supports for several features that exist in a standard J2SE VM have been omitted from the CLDC specification. The following list describes the features that are not supported in a CLDC-compliant VM. The features in this list have been omitted because of either changes to libraries or security concerns:

- Java Native Interface (JNI)
- user-defined class loaders
- reflection
- thread groups and thread daemons
- finalization weak references
- errors (a small subset of J2SE errors is supported)
- class file verification

Among these unsupported features, class file verification deserves further mention. The VM in the CLDC specification still performs this process, but it uses a two-step process and a different algorithm that requires fewer computation resources than the standard J2SE verifier.

The VM that comes with the CLDC reference implementation is called the Kilobyte Virtual Machine (KVM), so named because it uses only a few KB of runtime memory. It is a reference implementation that adheres to the CLDC specification's description of a compliant VM. The KVM is not a full-featured J2SE VM.

The specification of the features that a VM supports includes a specification of the libraries that it supports. The CLDC specification details the libraries that an implementation must support. As we know, a configuration is the basis for one or more profiles. The CLDC is a configuration on top of which one or more profiles are to be built in the same way that the Foundation Profile is built on top of the CDC. The intention is that the APIs in the CLDC profile support application development for the mass market of personal devices. The CLDC therefore targets third-party application developers. This is somewhat different than the CDC, which targets OEM developers.

Table.4 lists the packages that comprise the CLDC. The first three packages use the java.prefix in their name because each one contains a subset of the standard J2SE platform classes. The last one, however, must use the javax. prefix because it defines a new "standard extension" that is not part of the core Java platform.

Table.4 CLDC Packages

CLDC Package Name	Description
java.io	Standard Java IO classes and packages; subset of the J2SE package
java.lang	VM classes and interfaces; subset of the J2SE package
java.util	Standard utility classes and interfaces; subset of the J2SE package
javax.microedition.io	CLDC generic connection framework classes and interfaces

Because the category served by the CLDC encompasses so many different types of personal devices, potentially many different profiles are necessary to support them all. The most popular and well known of these is the Mobile Information Device Profile (MIDP), sometimes called the MID Profile [106]. The MIDP layers atop the CLDC and defines a set of user interface (UI) APIs designed for contemporary wireless devices.

Following in the tradition of Java parlance, MIDP applications are called MIDlets. A MIDlet is a Java application that uses the MIDP profile and the CLDC configuration.

The MIDP specification, like the CDC's Foundation Profile, was produced by an expert group, in this case, the Mobile Information Device Profile Expert Group, which is an international forum that includes representatives from several companies in the mobile device arena. The MIDP targets mobile information devices (MIDs), such as mobile phones, two-way pagers, and so forth, which have roughly the following characteristics [110]:

- screen size of approximately (at least) 96x54 pixels
- display depth of 1 bit
- one- or two-handed keyboard, touch screen input device
- 128 KB nonvolatile memory for MIDP components
- 8 KB nonvolatile memory for application-persistent data
- 32 KB volatile runtime memory for Java heap
- two-way wireless connectivity

Because the range of MID capabilities is so broad, the MIDP established a goal to address the least common denominator of device capabilities [107]. The MIDP, therefore, specifies the following APIs:

- application (MIDP application semantics and control)
- user interface
- persistent storage
- networking
- timers

Table.5 MIDP Packages	
MIDP Package Name	**Description**
javax.microedition.lcdui	UI classes and interfaces
javax.microedition.rms	Record management system (RMS) supporting persistent device storage
javax.microedition.midlet	MIDP application definition support class types
javax.microedition.io	MIDP generic connection framework classes and interfaces
java.io	Standard Java I/O classes and interfaces
java.lang	VM classes and interfaces
java.util	Standard utility classes and interfaces

A MIDP implementation must consist of the packages and classes specified in the MIDP specification. Additionally, it can have implementation-dependent classes for accessing native system software and hardware.

Figure.7 juxtaposes the CDC and CLDC platform stacks [108]. There is nothing inherent in either the CDC or CLDC that prohibits a manufacturer from porting either platform to a given family of devices. Nevertheless, the platform stacks specifically; the configuration and profile features have been specified to address practical limitations of the different families of hardware devices [109].

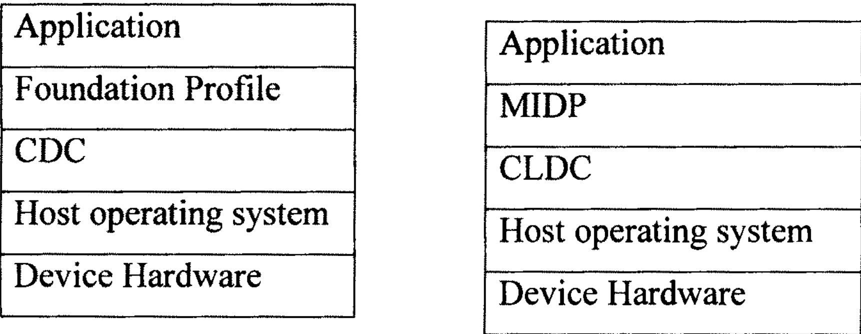


Figure.7 CDC and CLDC platform stacks

4.4.2. CARBIDE

Carbide Development Tools: Carbide is a new generation of mobile development tools from Nokia. More than just a new name, Carbide is a deliberate move to unify Nokia's mobile development tools into a common framework.

Carbide takes mobile development to a new level in terms of features and efficiency. Users now have one family for developing software for multiple platforms and multiple languages.Based on the open Eclipse framework, the Carbide offering can be extended with other eclipse plug-ins and products. Carbide tools will focus on three primary development areas:

Carbide Development Tools for Java - **Carbide.j** (formerly Nokia Developer's Suite for J2ME) is a software development tool for Java™ Platform, Micro Edition (Java™ ME) developers that enhances the development and verification of applications for Nokia devices. It provides tools for creating Mobile Information Device Profile (MIDP) and Personal Profile (PP) applications and deployment packages, signing applications, and deploying applications to devices. It is also an essential tool for managing, configuring, and running emulators for various Nokia platform and device SDKs.

Carbide Development Tools for Symbian OS C++: **Carbide.c++** is a family of Eclipse-based development tools supporting Symbian OS development on S60, Series 80, UIQ and MOAP.

Carbide Tools for personalization and customization of the user interface: **Carbide.ui** is a family of graphical WYSIWYG tools supporting Customization of mobile devices based on S60 and Series 40 platform [114]. The first product in this family is Carbide.ui S60 Theme Edition for Symbian OS.

Carbide.j is a tool for JavaTM Platform, Micro Edition (JavaTMME) developers that enhances the development and testing of applications for Nokia devices. It provides tools for creating mobile information device profile (MIDP) and Personal Profile (PP) applications and deployment packages, signing applications, and deploying applications to devices [115]. The latest version adds the ability to debug Java applications on S60 3rd Edition devices. Carbide.j is also an essential tool for managing, configuring, and running emulators for various Nokia platform and device SDKs.

Key Features of Carbide.J 1.5:

- Class creator.
- UI Designer.
- Screen Flow Designer.
- Web Services client tool.
- Package creator.
- Package signer.
- MIDP and PP support.
- DRM editor.
- Application deployer.
- Emulator management.
- On-device debugging for S60 3rd Edition devices.
- Apache Ant script support for tool features.

System Requirements:

- Microsoft Windows XP Professional (SP2 or later).
- 512 MB of RAM.
- 500 MB of disk space.
- 1-GHz or faster Pentium-class processor.
- Display capable of 16-bit color with 1,024 x 768 pixel resolution.
- Keyboard and mouse.
- Sun JavaTM Runtime Environment v 1.4.2_06 or later.

On-device debugging (ODD) :ODD is now supported for S60 3rd Edition devices, over Bluetooth and WLAN connections.

Updated Application Deployer : Nokia devices are now supported through integration with Nokia PC Suite 6.8.

Application wizards:Wizards help developers create new MIDP or PP applications, with the option to initialize the UI Designer, Game Designer and Screen Flow Designer according to the development need [117].

Create Class Tool: The Create Class Tool enables class source files to be created for a mobile Java application, speeding up development. Class source files are created using either the connected limited device configuration (CLDC) and MIDP API or connected device configuration (CDC) and PP API classes from the default a emulator device [116].

UI *design:* The UI Designer and Game Designer tools facilitate the design of MIDP user interface layouts and tiled layers for games. Support is provided for 176 x 208-pixel, 240 x 208-pixel, 352 x 416-pixel, 128 x 160-pixel, 240 x 320-pixel, and 208 x 208-pixel screens. The tools can be used with a stand-alone installation of Carbide.j or with a supported integrated development environment (IDE).

Screen Flow Designer: The Screen Flow Designer allows application logic to be easily created for MIDP UI designs. Using a drag-and-drop operation, the developer can move UI components into the designer view and then connect them together to create the application logic. The application code is generated automatically and can be tested easily in an emulator.

Web Services Client Tool: The Web Services Client Tool generates the stub classes for a Web services client from Web Services Description Language (WSDL) files. These classes provide a simple interface for accessing Web services and make it easier for developers to create mobile Web services applications.

Package creator: The Create Application Package tool assists with the construction of MIDP or PP application packages. It efficiently adds all the class and other files to the Java Archive (JAR) file and defines the Java application descriptor (JAD) file.

Package signer:The Sign Application Package tool is used to sign MIDP 2.0 MIDlet application packages [113]. Signing complies with the MIDP 2.0 security model using the signer's private key combined with the public key certificate. A signed application allows a device user to verify the sender and integrity of the application before installation.

Emulator management: Carbide.j manages the configuration and running of Nokia platform SDKs or Nokia device SDKs. This feature provides a single point from which to verify an application for many different devices.

Audio Converter:The Audio Converter tool converts MIDI or ring tone XML files to ring tone audio over-the-air (OTA) Java byte arrays and ring tone XML [111,112].

DRM Editor:Open Mobile Alliance (OMA) digital rights management (DRM) allows developers to control how a phone user uses and redistributes a MIDlet [104].The DRM editor can be used to protect MIDlets with a digital rights message.

Ant script support for tool features:Support is provided for several Carbide.j tool features, such as compilation, packaging, and execution, to be used in Ant scripts. Eclipse wizards also include automatic Ant build script creation.

IDE compatibility and related tools: Carbide.j is plug-and-play-ready for the following IDEs:

• Borland JBuilder 2005.

• Borland JBuilder X Mobile Edition.

• NetBeans IDE 4.1.

• NetBeans IDE 4.0.

• Eclipse 3.1.

• IBM WebSphere Studio Device Developer 5.7 (deployment tool only).

4.4.3. The Network Simulator Ns2

Introduction: Ns-2 is an object-oriented simulator developed as part of the VINT project at the University of California in Berkeley. Ns-2 is extensively used by the networking research community.

It provides substantial support for simulation of TCP, routing, multicast protocols over wired and wireless (local and satellite) networks, etc. The simulator is event-driven and runs in a non-realtime fashion. It consists of C++ core methods and uses **Tcl** and **Object Tcl** shell as interface allowing the input file (simulation script) to describe the model to simulate.

Users can define arbitrary network topologies composed of nodes, routers, links and shared media. A rich set of protocol objects can then be attached to nodes, usually as **agents**. The simulator suite also includes a graphical visualiser called network animator (**nam**) to assist the users get more insights about their simulation by visualising packet trace data.

It implements network protocols such as TCP and UPD, traffic source behavior such as FTP, Telnet, Web, CBR and VBR, router queue management mechanism such as Drop Tail, RED and CBQ, routing algorithms such as Dijkstra, and more. NS also implements multicasting and some of the MAC layer protocols for LAN simulations.

NS-2 includes a tool for viewing the simulation results, called NAM. NAM is a Tcl/TK based animation tool for viewing network simulation traces and real world packet trace data. The first step to use nam is to produce the trace file. The trace file should contain topology information, e.g., nodes, links, as well as packet traces. Usually, the trace file is generated by NS.

During an ns simulation, user can produce topology configurations, layout information, and packet traces using tracing events in ns. When the trace file is generated, it is ready to be animated by NAM. Upon startup, NAM will read the trace file, create topology, pop up a window, do layout if necessary, and then pause at the time of the first packet in the trace file. Through its user interface, NAM provides control over many aspects of animation.

Modeling in NS-2: To model a Network simulating using NS-2 is necessary to write a Tcl script describing the topology (nodes, agents, applications, etc.). In this chapter, we are going to define a very simple topology with two UDP nodes that are connected by a link and to send some data from node n0 to node n1. First, we have to create a simulator object. This is done with the command:

```
Set ns [new Simulator]
```

Now we open a file for writing that is going to be used for the nam trace data.

```
Set nf [open out.nam w]

$ns namtrace-all $nf
```

The first line opens the file 'out.nam' for writing and gives it the file handle 'nf'. In the second line we tell the simulator object that we created above to write all simulation data that is going to be relevant for nam into this file. The next step is to add a 'finish' procedure that closes the trace file and starts nam.

```
proc finish {} {
global ns nf
$ns flush-trace
close $nf

exec nam out.nam &
exit 0
}
```

The following two lines define the two network nodes.

```
set n0 [$ns node]
set n1 [$ns node]
```

A new node object is created with the command '$ns node'. The above code creates two nodes and assigns them to the handles 'n0' and 'n1'. The next line connects the two nodes.

```
$ns duplex-link $n0 $n1 1Mb 10ms DropTail
```

This line tells the simulator object to connect the nodes n0 and n1 with a duplex link with the bandwidth 1Megabit, a delay of 10ms and a DropTail queue.

```
#Create a UDP agent and attach it to node n0
set udp0 [new Agent/UDP]
$ns attach-agent $n0 $udp0
# Create a CBR traffic source and attach it to udp0
set cbr0 [new Application/Traffic/CBR]
$cbr0 set packetSize_ 500
$cbr0 set interval_ 0.005
$cbr0 attach-agent $udp0
```

The above lines create a UDP agent and attach it to the node n0, then attach a CBR traffic generator to the UDP agent. CBR stands for 'constant bit rate'. The packetSize is being set to 500 bytes and a packet will be sent every 0.005 seconds (i.e. 200 packets per second).

The next lines create a Null agent, which acts as traffic sink, and attach it to node n1.

set null0 [new Agent/Null]
$ns attach-agent $n1 $null0

Now the two agents have to be connected with each other.

$ns connect $udp0 $null0

Now we have to tell the CBR agent when to send data and when to stop sending (NS provides a very simple way to schedule events with the 'at' command.).

$ns at 0.5 "$cbr0 start"

$ns at 4.5 "$cbr0 stop"

The next line tells the simulator object to execute the 'finish' procedure after 5.0 seconds of simulation time.

$ns at 5.0 "finish"
The last line finally starts the simulation.
$ns run

Now it is possible to save the file (e.g. example1.tcl) and start the simulation.

Simulating a NS-2 network: After the example1.tcl simulation script is built, we run the simulation by executing the simulation script created in the previous section [118]. To run the simulation, type the following command:

ns example1.tcl

Starting NAM: To visualize the result it is possible to use a Network Animator (NAM). We can either start nam with the command 'nam <nam-file>' where '<nam-file>' is the name of a nam trace file that was generated by ns, or we can execute it directly out of the Tcl simulation script for the simulation which we want to visualize.Pressing on the 'play' button in the NAM window, we will see that after 0.5 simulation seconds, node 0 starts sending data packets to node 1.

CHAPTER 5

5. SIMULATION

Our algorithm is simulated using Carbide.j 1.5.for NOKIA S60 series mobiles for sending and receiving encrypted or decrypted SMS [120]. The same algorithm is executed in systems with the Dual Core and threading for sending or receiving encrypted or decrypted e-mails and instant messages also. In our experiments, we succeeded to transfer MIDlets from one device to another. This was possible thanks to software for Series 60 phones [119]. Then, all what remains to do is to choose "Options" and "send via Bluetooth", "SMS" or "Infrared." This operation is also possible in all Series 60 devices. These include Samsung, Siemens, Panasonic and mainly Nokia devices. [119]. finally, it is important to note that transferring is not possible for DRM.

5.1. SYSTEM SPECIFICATION

5.1.1. System Configuaration

Software Specifications:

Operating System - Windows XP With service pack 2

Browsers - Internet Explorer7.0

Java J2SE SDK 1.5.0_09

J2ME 1.4

Emulator:

CARBIDE.J 1.5 – NOKIA

- Series 60(S60), Series40 (S40)

Minimum Hardware Specifications:

- Server CPU Speed - Intel(r) Pentium(r)4 processor, 520, 2.8GHz, 1MB
- Cache, 800MHz FSB
- SINGLE processor/Dual core
- Storage Type Serial ATA
- Hard Drive Specs 7.2k RPM drives
- Hard Drive Space 72.8GB

Supported OS: This software is compatible with mobile phones having Symbian Operating System.

Installation options: This may be installed in either Phone memory or Card memory of the mobile phone depends on device design.

5.2. CHARTS

The following table and chart give the time taken for S60 mobiles and various processors to encrypt and send the message or non real time data.

Table 6: Simulation devices & their time Specifications

DEVICES	TIME (SECS)
Nokia N70	1.4327
Nokia 7710	1.6736
Intel Dual Core	0.0121
Intel Hyper Threading	0.0196
Laptops	0.60652
Pentium IV	0.000435
486 DX100	0.00248
Pentium III	0.000065

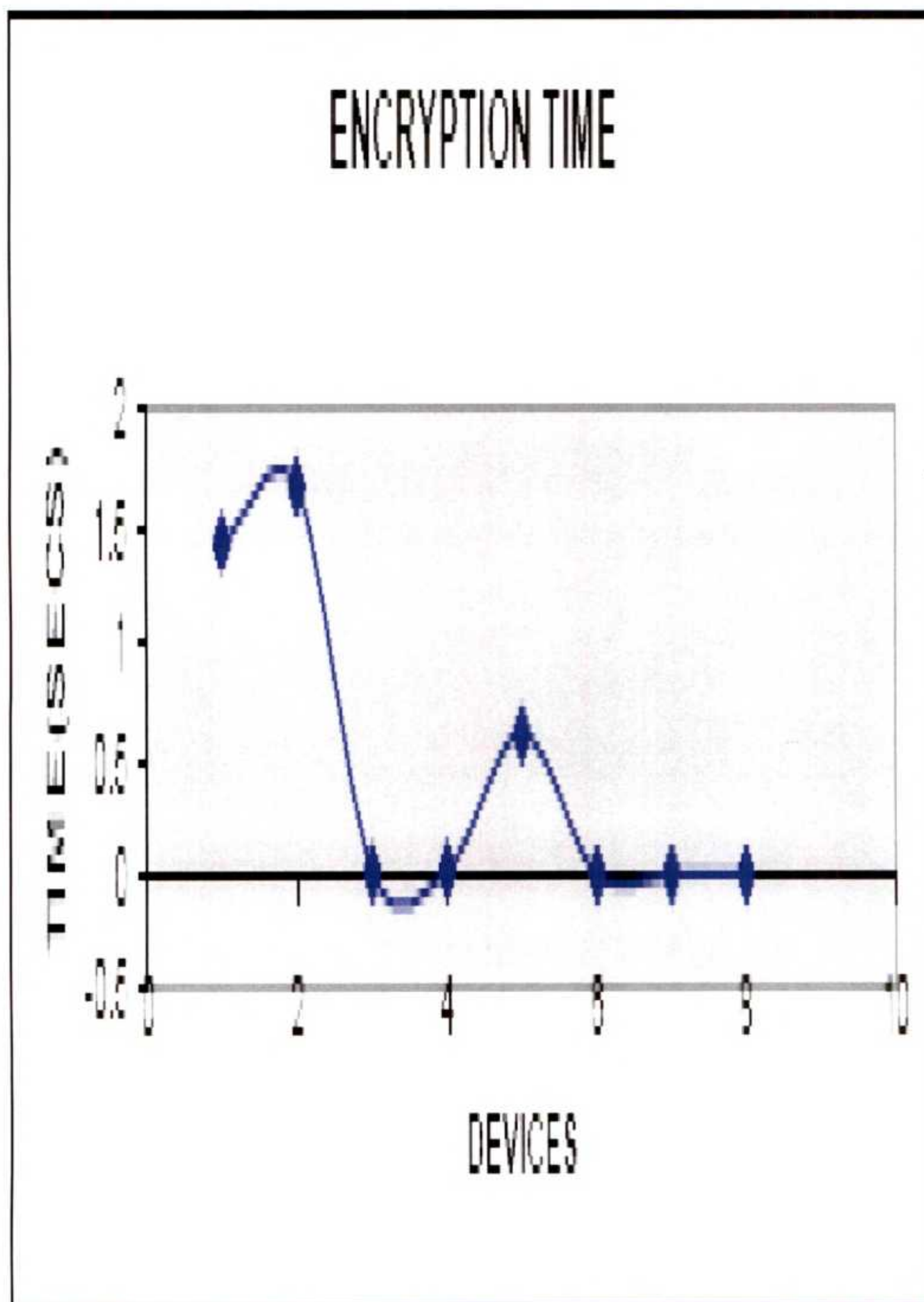


Figure.8 Relationships between the Simulation devices & their time slots

The next chart compares the time taken to hack the password by considering the samples of 250 bytes of messages with the other hashing algorithms.

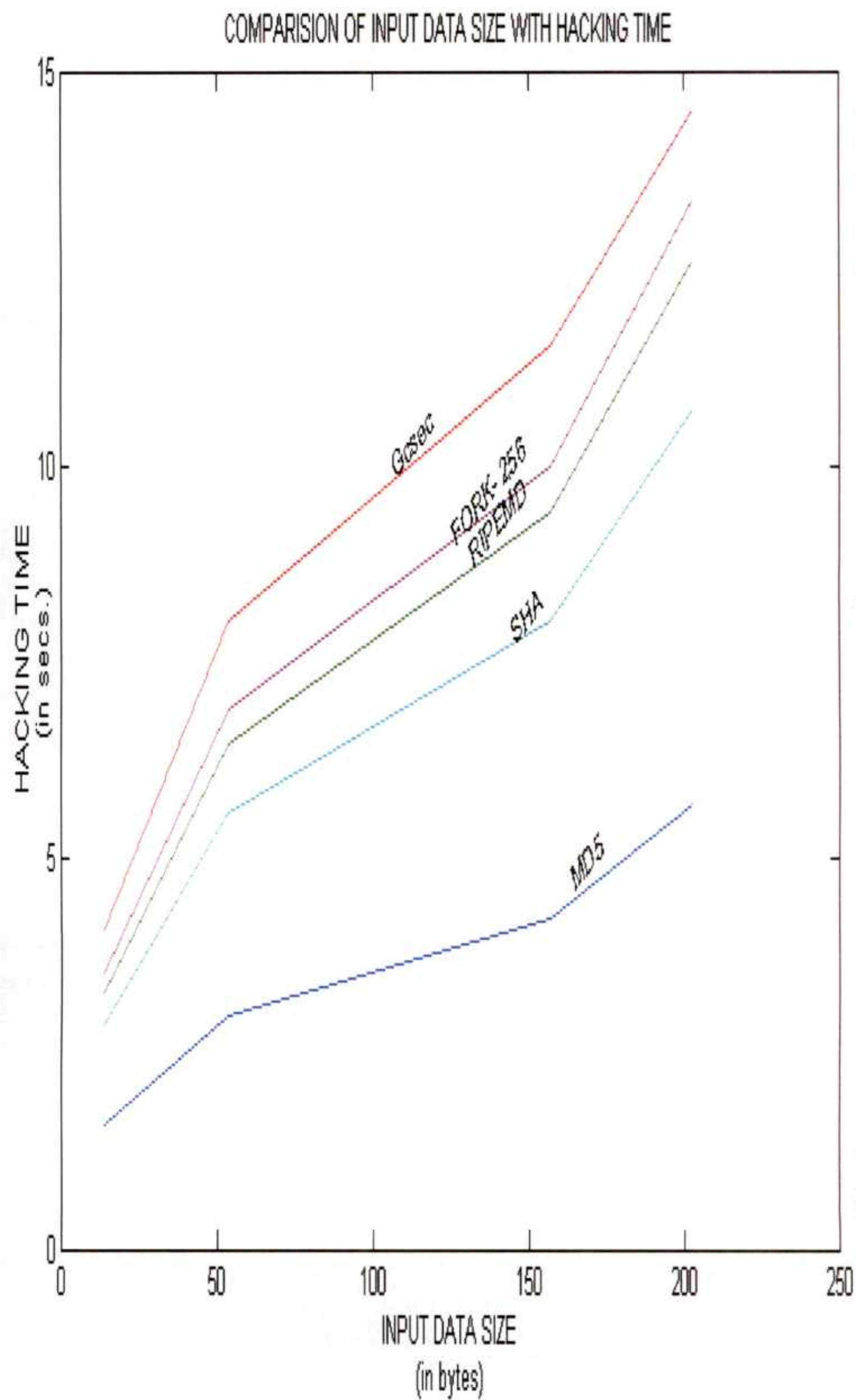


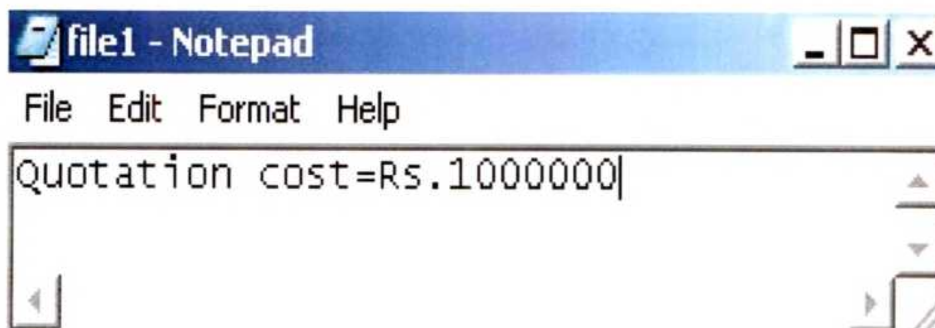
Figure.9 Comparisons of input with hacking time

5.3. SIMULATIONS OF VARIOUS DATA

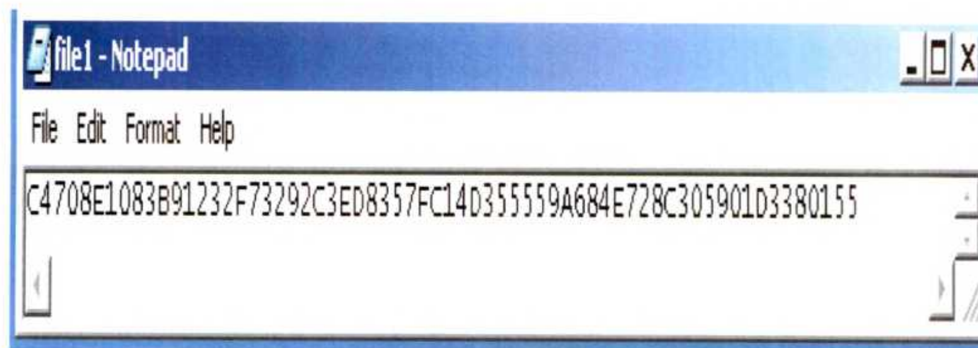
5.3.1. E-MAIL

This algorithm is simulated in J2ME and analyzed in MATLAB 6.1 for computer and mobile networks to encrypt and decrypt the E-mail .

INPUT



OUTPUT



5.3.2. SMS/Instant Messages

Carbide.j provides tools for creating and packaging Mobile Information Device Profile (MIDP) applications, and also provides a convenient interface for managing Nokia Java SDKs. These SDKs facilitate application testing on a PC, without the need for a device, with emulators for the Series 40 Platform, the S60 Platform, the Series 80 Platform, and the Nokia 7710 widescreen multimedia smart phone.

We have to specify the recipient name and phone number before typing the message in the screen or chat room for SMS and instant messages respectively which is shown in figure.10. The sender receives these with the message and passed as argument to the software to encrypt .The system waits for the confirmation of user trustiness for sending the message as encrypted message.

The figure 11 shows the original message of 450 characters at the maximum what to be sent. The figure 12 shows all the Classes and attributes to which the program or procedure to be executed to create CLDP and MIDIlet which is to be simulated. Then the fig.13 shows that the packet of data sent contains the address information of the recipient, the data and an address header identifying us to the recipient.

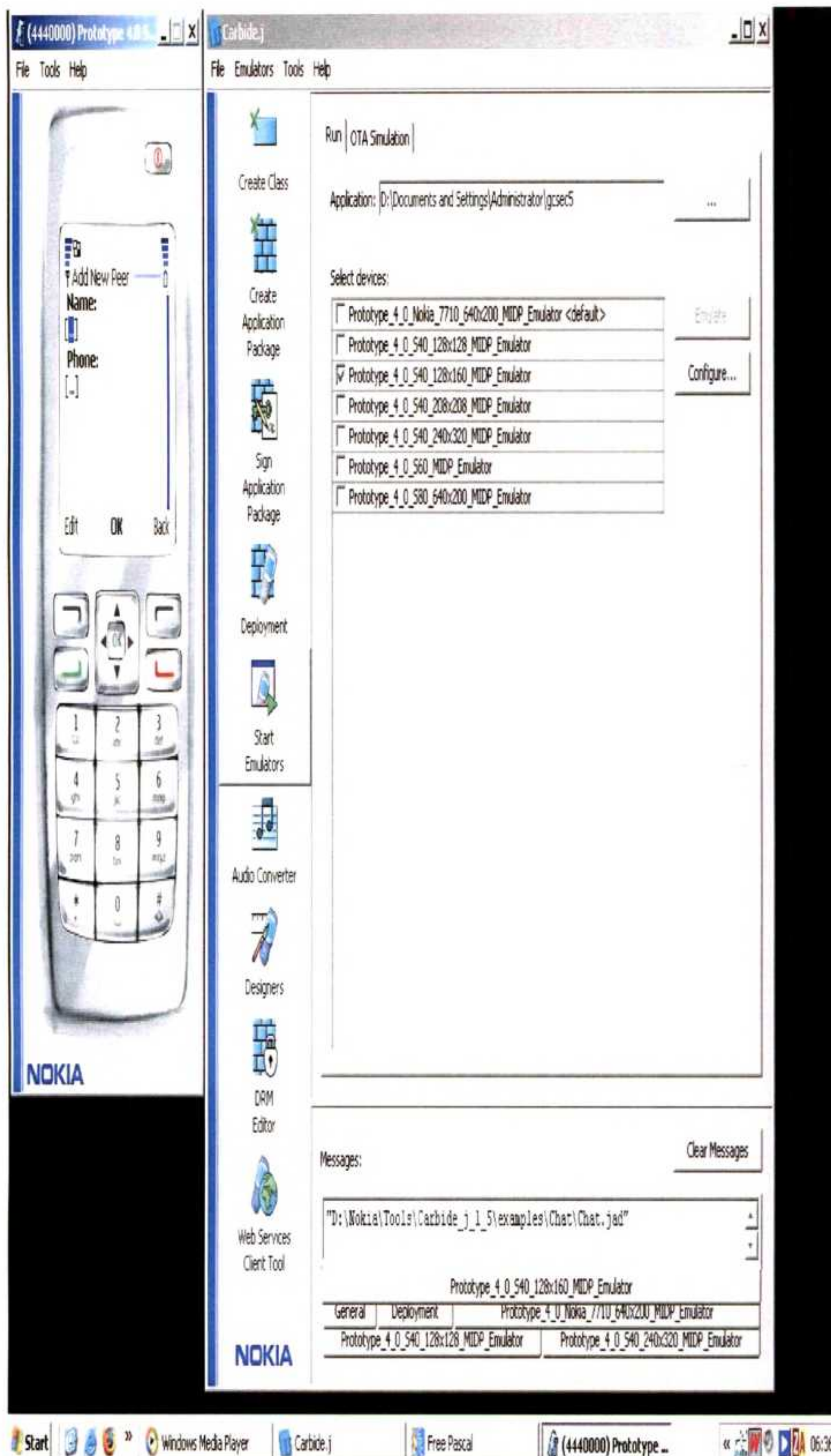


Figure.10 SMS input screen

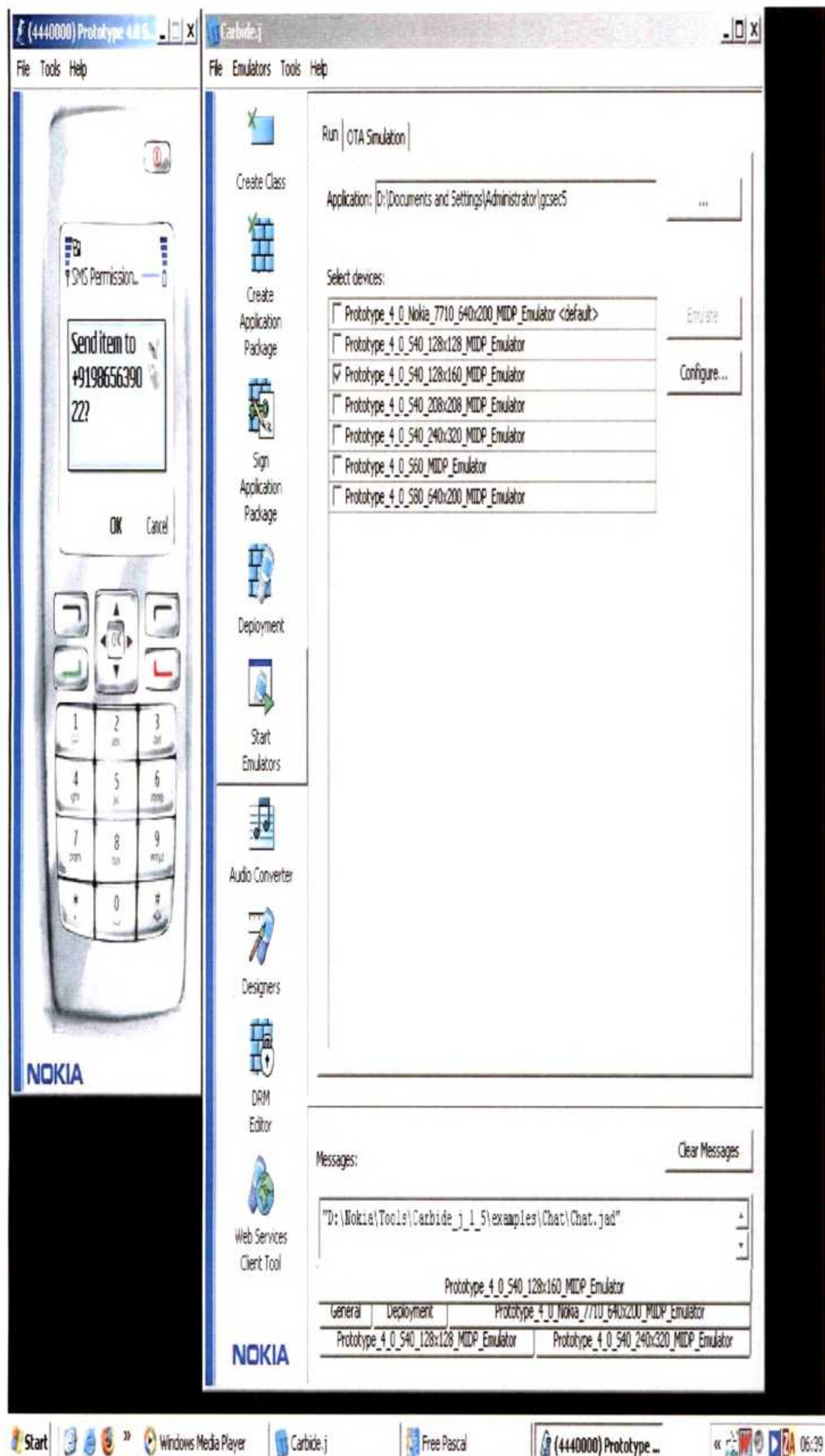


Figure.11 Destination specification & Message

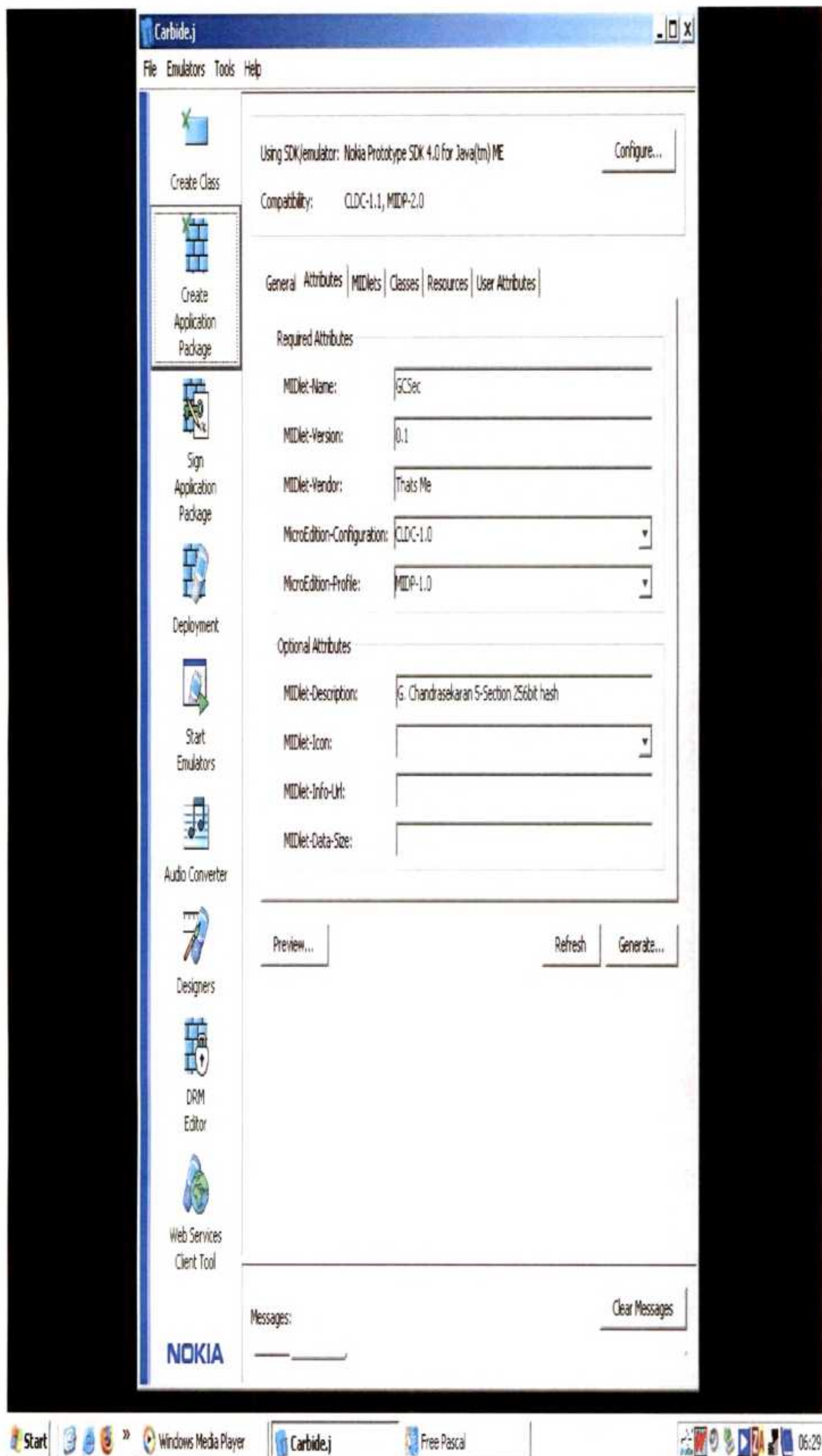


Figure.12 Classes & attributes in CLDC

The receiver receives the encrypted message as either normal or Bluetooth message which is to be decrypted here. If a user wants to store the message as encrypted message itself in inbox, then the receiver software has to be deactivated then the resultant message will be in the form of encrypted alphanumeric characters.

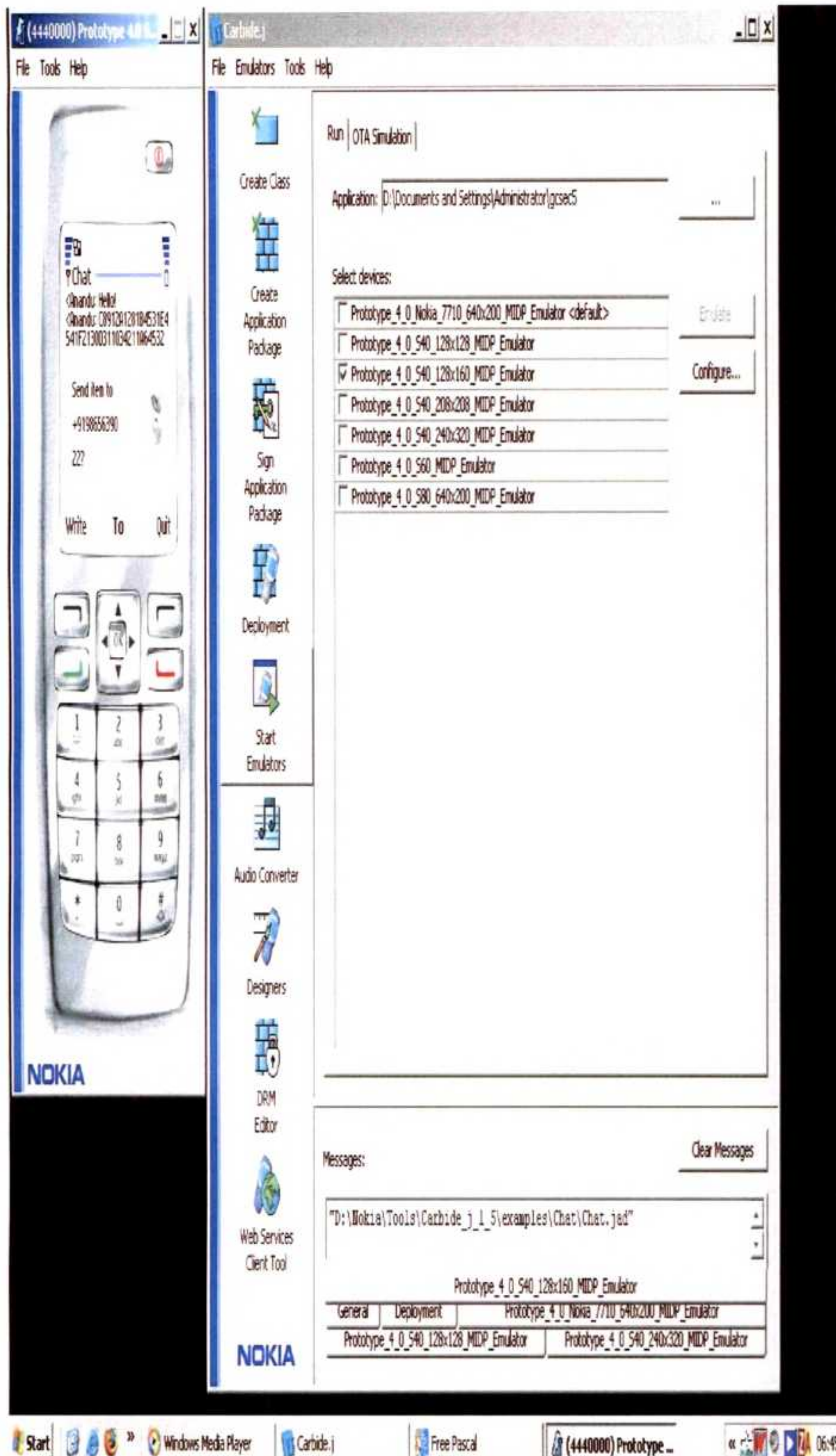


Figure.13 Encrypted result

CHAPTER 6

CONCLUSION

Our algorithm is simulated for sending or receiving non real time data in mobiles of S60 series, NOKIA 7710 & N series and various system processors with less transmission delay. Since the complexity of this algorithm is increased by introducing many number of steps for shifting the left and right parts of the bits of the data and complex XOR operations, GCSEC is having the highest level of privacy also which introduces the less probability of even assuming the password .

FUTURE ENHANCEMENTS

1. Extending our algorithm for server based model.
2. Our algorithm works for S60 series of Nokia that can be generalized for all types of mobiles.

REFERENCES

1. Stallings W, Data and Computer Communications, sixth edition, Prentice Hall, Upper Saddle River, N.J., 2000.

2. Comer DE, Computer Networks and Internets, second edition, Prentice Hall, Upper Saddle River, N.J., 1999.

3. Halsall F, Data Communications, Computer Networks and Open Systems, fourth edition, Addison-Wesley, Reading, Mass., 1996.

4. Leon-Garcia A &Widjaja I, Communication Networks: Fundamental Concepts & Key Architectures, McGraw-Hill Higher Education, New York, 2000.

5. Garg VK & Wilkes JE, Wireless and Personal Communications Systems, Prentice Hall, Upper Saddle River, N.J., 1996.

6. Jamalipour A, Low Earth Orbital Satellites for Personal Communication Networks, Artech House Publishers, Norwood, Mass., 1998.

7. "Digital cellular technologies", Special Issue of IEEE Transactions on Vehicular Technologies, 40(2), 1991.

8. Abramson N, "The ALOHA system—another alternative for computer communications", Proceedings 1970 Fall Joint Computer Conference, 1970, pp. 281–285.

9. Viterbi AJ, "CDMA—Principles of Spread Spectrum Communications", Addison-Wesley, Reading, Mass., 1995.

10. Gilhousen KS, Jacobs IM, Padovani R, Viterbi AJ, Weaver LA & Wheathley III CE, "On the capacity of a cellular CDMA system", IEEE Transactions on Vehicular Technologies,40(2), 303–312, 1991.

11. Prasad .R, "CDMA for Wireless Personal Communications", Artech House Publishers, Norwood, Mass., 1996.

12. Prasad R & Ojanpera T, "An overview of CDMA evolution toward wideband CDMA", IEEE Communications Surveys,**1**(1),FourthQuarter,1998,http://www.comsoc.org/pubs/surveys.

13. Wilson ND, Ganesh R, Joseph K & Raychaudhuri D, "Packet CDMA versus dynamic TDMA for multiple access in an integrated voice/data PCN", IEEE Journal on Selected Areas in Communications, **11**(6), 870–884, 1993.

[14] Cardwell, A., and Woollard, S. "Clinic: What is the biggest security risks associated with wireless technology? What do I need to consider if my organization wants to introduce this kind of technology to my corporate LAN?" www.itsecurity.com, 2001.

[15] Marek, S. "Identifying the Weakest Link." Wireless InternetMagazine,www.wirelessinternetmag.com,November/December, 2001.

[16] Rene Meier, "Common paradigms of mobile computing", Mobile Computing and Communications Review,Volume6(4),https://www.cs.tcd.ie/publications/tech-reports/reports.03/TCD-CS-2003-23.pdf,2003.

[17] Attewell J, Savill-Smith C (eds) (2004). "Learning with mobile devices: research and development –a book of papers". London: Learning and Skills Development Agency. www.LSDA.org.uk/.les/PDF/1440.pdf, accessed July 2004.

[18] van Grinsven L, "Lost? Your phone knows a way out", USAToday,August2004.

[19] H. Schotten, "Evolution of 3G radio access techniques", in Proc. of the International Symposium '3G Infrastructure and Services' 3GIS, Athens, pp. 161-165, 2001.

[20] Y. Guo and H. Chaskar, "Class-based quality of service over air interfaces in 4G mobile networks", IEEE Commun. Mag., pages 132-137, March 2002.

[21] David Bollier, rapporteur, "When Push Comes to Pull: The New Economy and Culture of Networking Technology" 2005.

[22] B.G. Evans and K. Baughan, "Visions of 4G", Electronics & Communication Engineering Journal, Vol. 12, No. 6, pp.2,2004.

[23]. Cai J & Goodman DJ, "General packet radio service in GSM", IEEE Communications Magazine, **35**(10), 122–131, 1997.

[24]. Brasche G & Walke B, "Concepts, services, and protocols of the new GSM phase2+ general packet radio service", IEEE Communications Magazine, **35**(10), 94–104, 1997.

[25]. Bettstetter C, Vogel H-J & Eberspacher J, "GSM phase 2+ general packet radio service GPRS: architecture, protocols, and air interface", IEEE Communications Surveys, **2**(3), 1999, http://www.comsoc.org/pubs/surveys.

[26]. Rahnema M, "Overview of the GSM system and protocol architecture", IEEE Communications Magazine, **31** (4), 92–100, 1993.

[27]. Mobile Wireless Internet Forum (MWIF), http://www.mwif.org.,2000.

[28]. Umehira M, Nakura M, Umeuchi M, Murayama J, Murai T & Hara H, "Wireless and IP integrated system architectures for broadband mobile multimedia services", Proceedings of IEEE Wireless Communications and Networking Conference (WCNC '99), New Orleans, 1999.

[29]. Macker JP, Park VD & Corson MS, "Mobile and wireless Internet services: putting the pieces together", IEEE Communications Magazine, June, 148–155, 2001.

[30]. Oliphant MW, “The mobile phone meets the Internet”, IEEE Spectrum, August, 20–28, 1998.

[31]. Noerenberg II JW, “Bridging wireless protocols”, IEEE Communications Magazine, November, 90–97, 2001.

[32]. McCann PJ, Hiller T, “An Internet infrastructure for cellular CDMA networks using mobile IP”, IEEE Personal Communications Magazine, August, 6–12, 2000.

[33]. Ramjee R, La Porta TF, Thuel S & Varadhan K,”IP-based access network architecture for next-generation wireless data networks”, IEEE Personal Communications Magazine, August, 34–41, 2000.

[34] Jill Attewell, Giorgio Da Bormida, Mike Sharples and Carol Savill-Smith, “MLEARN 2003 learning with mobile devices,” www.LSDA.org.uk/events/mlearn,2003.

[35] Dave Singelee and Bart Preneel, “The Wireless Application Protocol”, International Journal of Network Security, Vol.1, No.3, PP.161–165, Nov. 2005.

[36] “Java 2 Platform Micro Edition – J2ME”, Official Site, http://java.sun.com/javame/index.jsp.

[37] AvantGo Security White Paper “Ensuring Mobile Security With AvantGo Technology”, 2002.

[38] ABADI, M., BURROWS, M., KAUFMAN, C., AND LAMPSON, B. "Authentication and delegation with smart-cards. In Theoretical Aspects of Computer Software", LNCS 526, Springer, 1991, pp. 326-345. Also Res. Rep. 67, Systems Research Center, Digital Equipment Corp., Palo Alto, Calif., Oct. 1990.

[39] DAVIS, D. AND SWICK, R.,"Network security via private-key certificates". ACM Oper. Syst.Rev. 24, 4 (Oct. 1990), 64-67.

[40] Judith Barnes/Crawford Warnock at Citigate Dewe Rogerson, "Spam back as top email security threat Junk email bigger problem than viruses", 2006,www.bsg.co.uk/newsevents/pdfs/spam_threat_release.pdf.

[41] H. Kikuchi, M. Tada, and S. Nakanishi, "Secure instant messaging protocol preserving confidentiality against administrator," in 18th International Conference on Advanced Information Networking and Applications, AINA 2004, vol. 2, Fukuoka, Japan, Mar. 2004, pp. 27–30.

[42] N. Hindocha, "Threats to instant messaging," http://securityresponse.symantec.com/avcenter/reference/threats.to.instant.messaging.pdf,2003.

[43] M. D. Murphy, "Instant message security - Analysis of Cerulean Studios Trillian application," http://www.giac.org/practical/GSEC/Michael Murphy GSEC.pdf , June 2003.

[44] D. Frase, "The instant message menace: Security problems in the enterprise and some solutions," Institute, http://www.sans.org/rr/papers/60/479.pdf, Nov. 2001.

[45] Short Message Service/SMS tutorial. http://www.developers home.com/sms.

[46] TIA/EIA-637-A, Short Message Service. Telecommunications Industry Association, December1999.

[47] G. Peersman, S. Cvetkovic, P. Griffiths, and H. Spear.
"The global system for mobile communications Short Message Service", IEEE Personal Communications, June 2000.

[48] 3GPP. TS 23.040, "Technical Realization of the Short Message Service (SMS)", Release 6. v6.5.0, September 2004.

[49] 3GPP. TS 22.140, "Multimedia messaging service release 6. v6.7.0, March 2005.

[50] 3GPP2. S.r0061, "wireless immediate messaging. v1.0", October 2002.

[51] Y. Omori, T. Suda, G. Lin, and Y. Kosugi, "Feedback-based congestion control for VBR video in ATM networks," in 6th Int. Workshop Packet Video'94, Dallas, TX, Sept. 1994.

[52] C. M. Sharon, M. Devetsikiotis, I. Lambadaris, and A. R. Kaye, "Rate control of VBR H.261 video on frame relay networks," in IEEE ICC'95, June 1995.

[53] H. Kanakia, P. P. Mishra, and A. Reibman, "An adaptive congestion control scheme for real-time packet video transport," IEEE/ACM Trans.Networking, vol. 3, Dec. 1996.

[54] B. J. Vickers, M. Lee, and T. Suda, "Feedback control mechanisms for real-time multipoint video services," IEEE J. Select. Areas Commun.,vol. 15, Apr. 1997.

[55] A. Bar-Noy and I. Kessler, "Tracking mobile users in wireless communications networks," in IEEE INFOCOM'93, Mar.1993.

[56] A. Bar-Noy, I. Kessler, and M. Sidi, "To update or not to update?," in IEEE INFOCOM'94, June 1994.

[57] U. Madhow, M. L. Honig, and K. Steiglitz, "Optimization of wireless resources for personal communications mobility tracking," in IEEE INFOCOM'94, June 1994.

[58] D. Raychaudhuri and N. D. Wilson, "ATM- based transport architecture for multi services wireless personal communications networks," IEEE J. Select. Areas Commun., vol. 12, Oct. 1994.

[59] V. Parson, "Empirically derived analytic models of wide-area TCP connections," IEEE/ACM Trans. Networking, vol. 2, Aug. 1994.

[60] K. M. S. Murthy and R. Pandya, "Tutorial on personal communication systems and services," in IEEE ICUPC'94, San Diego, CA, Sept. 1994.

[61] Z. Haas, "Tutorial on mobile communication networks," in IEEE GLOBECOM'94, Nov. 1994.

[62] J. Crowcroft, S. Hailes, M. Handley, A. Jena, D. Lewis, and I. Wakeman, "Some multimedia traffic characterization and measurement results," U.K., 1992.

[63] L. Munoz, M. Garcia, and J. Zamanillo, "Proposal of a corporate network architecture for a mobile architecture for a mobile digital system of voice and data communications," IEEE ICUPC'94, San Diego, CA,Sept. 1994.

[64] Carlos Oliveira, Jaime Bae Kim, "An Adaptive Bandwidth Reservation Scheme for High-Speed Multimedia Wireless Networks", IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, VOL. 16(6), AUGUST 1998.

[65] White paper, "GSM SIM with SECURE TEXT MESSAGING", http://www.3gforensics.co.uk/index.html, 2006.

[66] Jonathan Knudsen,"MIDP Application Security-EncryptioninMIDP",http://developers.sun.com/mobility/midp/articles/security4, 2005.

[67] CryptoSMS – "Protecting your confidential SMS messages",http://www.cryptosms.com/protect.html ,2006.

[68]UniLeadtoneLimited,http://www.ming.host.sk/index.php-action/ contact.html, 2003.

[69]Easyhelper,http://www.easyhelper.net/smssecurity.html, 2007.

[70]CryptoGraf,http://www.allaboutsymbian.com/software/item/CryptoGraf_Messaging_v20_S60_3rd_Ed.php, "CryptoGraf_Messaging" 2006.

[71]Best_Jotter, "Security software for SMS" http://www.allabout symbian.com/software/item/Best_Jotter_for_S60_3rd_edition.php, 2006.

[72]Kryptext, "SMS cryptography with privacy software", http://www.allaboutsymbian.com/software/item/Kryptext.php, October 7th 2004.

[73]circletech.net, "Enjoy security of your SMS "http://www.circletech.net/download/en/press03062006.pdf, 2006.

[74] Nokia white paper, "SMS security tools", http://www.softwaremarket.nokia.com/images/EN/user_guide_1951.pdf, SMS 007 system, Sept 12, 2005.

[75] NSS MSC Sdn Bhd (624307-K), Suite E-07-21, Plaza Mont' Kiara, "Network Security Solutions", XMS Manager 2005.

[76] Solidlabs white paper,"Encrypter for chatting", http://solidlabs.com/chatencrypter, June 2004.

[77] Donald J. Longueuil. "Wireless Messaging Demystified: SMS, EMS, MMS, IM, and others", McGraw-Hill Professional Publishing, 2003.

[78] E. Biham and R. Chen, "Near-Collisions of SHA-0," Advances in Cryptology –CRYPTO 2004, LNCS 3152, Springer-Verlag, pp. 290–305, 2004.

[79] E. Biham, R. Chen, A. Joux, P. Carribault, C. Lemuet and W. Jalby, "Collisions of SHA-0 and Reduced SHA-1," Advances in Cryptology – EUROCRYPT 2005, LNCS 3494, Springer-Verlag, pp. 36–57, 2005.

[80] B. den Boer and A. Bosselaers, "An Attack on the Last Two Rounds of MD4," Advances in Cryptology – CRYPTO'91, LNCS 576, Springer-Verlag, pp. 194–203,1992.

[81] B. den Boer and A. Bosselaers, "Collisions for the Compression Function of MD5,"Advances in Cryptology – CRYPTO'93, LNCS 765, Springer-Verlag, pp. 293–304,1994.

[82] F. Chabaud and A. Joux, "Differential Collisions in SHA-0," Advances in Cryptology– CRYPTO'98, LNCS 1462, Springer-Verlag, pp. 56–71, 1998.

[83] I. Damgard, "A Design Priciple for Hash Functions," Advances in Cryptology –CRYPTO'89, LNCS 435, Springer-Verlag, pp. 416–427, 1989.

[84] H. Dobbertin, "RIPEMD with Two-Round Compress Function is Not Collision-Free," Journal of Cryptology 10:1, pp. 51–70, 1997.

[85] H. Dobbertin, "Cryptanalysis of MD4," Journal of Cryptology 11:4, pp. 253–271,1998.

[86] H. Dobbertin, A. Bosselaers and B. Preneel, "RIPEMD-160, a strengthened version of RIPEMD," FSE'96, LNCS 1039, Springer-Verlag, pp. 71–82, 1996.

[87] R. C. Merkle, "One way hash functions and DES," Advances in Cryptology –CRYPTO'89, LNCS 435, Springer-Verlag, pages 428–446, 1989.

[88] NIST/NSA, "FIPS 180-2: Secure Hash Standard (SHS)", August 2002 .

[89] R. L. Rivest, "The MD4 Message Digest Algorithm," Advances in .Cryptology –CRYPTO'90, LNCS 537, Springer-Verlag, pp. 303–311, 1991.

[90] R. L. Rivest, "The MD5 Message-Digest Algorithm," IETF Request for Comments,RFC 1321, April 1992.

[91] B. Van Rompay, A. Biryukov, B. Preneel and J. Vandewalle, "Cryptanalysis of 3-pass HAVAL," Advances in Cryptology – ASIACRYPT 2003, LNCS 2894, Springer-Verlag, pp. 228–245, 2003.

[92] X. Wang, X. Lai, D. Feng, H. Chen and X. Yu, “Cryptanalysis of the Hash Functions MD4 and RIPEMD,” Advances in Cryptology – EUROCRYPT 2005, LNCS 3494, Springer-Verlag, pp. 1–18, 2005.

[93] X. Wang and H. Yu, “How to Break MD5 and Other Hash Functions,” Advances in Cryptology – EUROCRYPT 2005, LNCS 3494, Springer-Verlag, pp. 19–35, 2005.

[94] X. Wang, H. Yu and Y. L. Yin, “Efficient Collision Search Attacks on SHA-0,” Advances in Cryptology – CRYPTO 2005, LNCS 3621, Springer-Verlag, pp. 1–16,2005.

[95] X. Wang, Y. L. Yin and H. Yu, “Finding Collisions in the Full SHA-1,” Advances in Cryptology – CRYPTO 2005, LNCS 3621, Springer-Verlag, pp. 17-36, 2005.

[96] Y. Zheng, J. Pieprzyk and J. Seberry, “HAVAL – A One-Way Hashing Algorithm with Variable Length of Output,” Advances in Cryptology – AUSCRYPT’92, LNCS 718, Springer-Verlag, pp. 83–104, 1993.

[97] Deukjo Hong, Jaechul Sung, Seokhie Hong, Sangjin Lee, and Dukjae Moon, ”A New Dedicated 256-bit Hash Function”,csrc.nist.gov/groups/ST/hash/documents/Sung_FORK-256.pdf,2006.

[98]. RIVEST, R. “The MD4 message digest algorithm”, In Advances in Cryptology-----Crypto ’90, Springer, 1991, pp. 303-311.

[99]. RIVEST, R. AND DUSSE, S.,"The MD5 Message-Digest Algorithm", Internet Draft [MD5-A]:draft-rsadsi-rivest-md5-01.txt, July 1991.

[100]. Access Data corp. white paper, "Password Recovery with PRTK™/DNA", March, 2006,. http://access data.com/media /en_US/print/papers/wp.PRTKDNA_Password_Recovery.en_us.pdf.

[101]. Mourad debbabi, Mohamed Saleh, Chamseddine Talhi and Sami Zhioua , "Java for Mobile Devices: A SecurityStudy",ieeexplore.ieee.org/iel5/10467/33214/01565251.pdf,2006.

[102] Developnet, "Why is J2ME MIDP superior to WAP", http://www.developnet.co.uk/wap.htm,February 25, 2007.

[103] Phones,Yesim Tunccekic and Kivanc Dincer "Mobile Mapping Applications over J2ME Enabled", IJCSNS International Journal of Computer Science and Network Security, VOL.7 No.2, February 2007,pp:316-322.

[104]. Nokia white paper,"Implementation Best Practices for OMA DRM v1.0 Protected MIDlets", May 2004.

[105] Nokia official site paper:,"CLDC 1.0 (JSR-30)", http://jcp.org/en/jsr/detail?id=30 and CLDC 1.1 (JSR-139) .

[106] "Java MIDP Application Developer's Guide for Nokia Devices v1.0", http://www.forum.nokia.com/java.

[107] "Efficient MIDP Programming v1.1", http://www.forum.nokia.com/info/sw.nokia.com/id/d307878fbbd6415aaf25bf7fb3efc9d3/Efficient_MIDPProgramming_v1_1_en.pdf.html.

[108] "Designing MIDP Applications for Optimization", http://www.forum.nokia.com/info/sw.nokia.com/id/ff51fcc6-edc0763987489527700c7ff/Designing_MIDP_Applications_For_Optimization_v1_0_en.pdf.html.

[109] J. V. Peursem.,"JSR 118 Mobile Information Device Profile 2.0", November 2002.

[110] Inc. Sun Microsystem, "Mobile Information Device Profile(MIDP)Specification"http://java.sun.com/products/midp/ [2000] 93-303, Dec. 2000.

[111] "OTA download for generic content – Introduction", http://www.forum.nokia.com/info/sw.nokia.com/id/685dddec-38c8-419d-aaca-3639b8f03705/COD_intro.pdf.html.

[112] "Mobile Information Device Profile 2.0 (JSR-118)", http://jcp.org/en/jsr/detail?id=118.

[113]"Settings for OTA Download of MIDlets v1.0", http://www.forum.nokia.com/info/sw.nokia.com/id/9eb5643d433a4ee2956bed0f5fb6ced9/Settings_for_OTA_Download_of_MIDlets_v1_0.pdf.html.

[114] "List of developement tools for JavaME", http://www.j2meforums.com/wiki/index.php/Development_Tools.

[115] Yesim Tunccekic and Kivanc Dincer ,"Mobile Mapping Applications over J2ME Enabled Phones", IJCSNS International Journal of Computer Science and Network Security, VOL.7 No.2, February 2007,pp:316-322.

[116] Carbide.j,"Eclipse plugin for development of JavaME",Symbianhttp://forum.nokia.com/info/sw.nokia.com/id/d9f7e9b2-3932-4358-9e8e-aa5cd26be54e.html.

[117]Simon Judge,"Archive for the tools", 2006, http://www.mobile phone development.com/ archives/category/ tools/carbide.j.

[118]MarcGreis, "Tutorial for ns2",www.ns-tutorial/ns2.2.6.

[119]Nokia.Series60Platform,http://www.nokia.com/nokia/0,8764,46827,00.html.

[120]"Symbian OS & SDKs,http://developer.symbian.com/main/tools.

PUBLICATIONS RELATED WITH THIS RESEARCH

1. INTERNATIONAL JOURNALS

1. M.Ganaga durga, G.Chandrasekaran, S.Arivazhagan, "Security (GCSEC) algorithm for non real time data in mobile networks", Journal of Mobile Communication, Vol.1, No.4, pp 109-113, 2007.

2. M.Ganaga durga, G.Chandrasekaran, S.Arivazhagan, "Hash Cryptography for SMS banking", communicated to the International Journal of Computer Systems Science and Engineering .

2. NATIONAL JOURNALS

1. M.Ganaga durga, G.Chandrasekaran, S.Arivazhagan, "E-mail security", communicated to the Journal of Mathematics & Cryptography, New Delhi.

2. M.Ganaga durga, G.Chandrasekaran, S.Arivazhagan, "Security algorithm for e-mail transaction in Mobile networks", communicated to the Journal of IETE Technical Review (IETE), New Delhi.

3. INTERNATIONAL CONFERENCES

1. M.Ganaga durga, G.Chandrasekaran, S.Arivazhagan, "QOS to secure non real time data", Published in the Proceeding of the International Conference on Communication & Power Systems (ICCPS'06), pp-215-219, 2006.

2. M.Ganaga durga, G.Chandrasekaran, S.Arivazhagan, "Dedicated 256 hash algorithm for non real time data", Published in the Proceeding of the International Conference on Communication & Power Systems (ICCPS'06), pp-198-202, 2006.

3.M.Ganaga durga, G.Chandrasekaran, S.Arivazhagan, "Hash Cryptography for SMS banking", Published in the Proceeding of the International Conference on Advances in Electronics & Communications (icon ADELCO 2007),pp-133-138, 2007.

4.M.Ganaga durga, G.Chandrasekaran, S.Arivazhagan, "An Authentication & Security Protocol For Mobile Computing", Published in the Proceeding of the International Conference on Advanced Computing & Communications (ICCACC 2007), pp-582-585, 2007.

5. M.Ganaga durga, G.Chandrasekaran, S.Arivazhagan, "CIPHER TALK-Encryption algorithm for mobile networks", Published in the Proceedings of the International Conference on Computational Intelligence and Multimedia Applications (ICCIMA 2007), Vol.4 , pp:363-367 and also published in IEEEXplore, December 2007.

4. NATIONAL CONFERENCES

1. M.Ganaga durga, N.Ramaraj,"Mobile Communication- Adaptive Bandwidth Reservation Schemes", Published in the Proceeding of the National Conference on Computer communication & mobile computing systems (NCCC-MCS), pp-2, 2005.

2. M.Ganaga durga, N.Ramaraj,"Mobile Communication- Qos To Secure Hand-Offs", Published in the Proceeding of the National Conference on Advances in Electronics & communications (icon ADELCO 2005),pp-368-373, 2005.

COPIES OF PUBLICATIONS

"Security (GCSEC) algorithm for non real time data in mobile networks", Journal of Mobile Communication, Vol.1, No.4, pp 109-113, 2007.

Journal of Mobile Communication 1 (4): 109-113, 2007
ISSN: 1990-794X

Security (GCSEC) Algorithm for non Real Time Data in Mobile Networks

[1]M. Ganaga Durga, [2]G. Chandrasekaran and [3]S. Arivazhagan
[1]Department of M.C.A., K.L.N. College of Engineering, Tamilnadu, India
[2]Department of Computer Applications and [3]Department of ECE,
Mepco Schlenk Engineering Collge, Tamilnadu, India

Abstract: The focal hub delves into conniving an proficient 256-bit hash utility to shield the non real time data and its hacking echelon is compared with the hash algorithms. This study investigates the recital of hashing algorithms by an untried loom. Hash functions have ample and imperative role in cryptography. They generate hash ethics, which succinctly embody longer messages or credentials from which they were computed. The focal task of cryptographic hash functions is in the stipulation of message veracity checks and digital signatures. This algorithm utterly secures the non real time data by encrypting the data what is sent or time-honored.

Key words: Refuge, hash utility, psychiatry, hacking tempo, digest

INTRODUCTION

Security-an ever-growing concern: Security requirements in any transaction are directly proportional to the transaction's value, sensitivity and volume. Mobile transactions in the cash management industry typically possess all these characteristics. Indeed, security is a *de facto* requirement for any transaction channel in the cash management business andthus most security requirements are fairly generic and independent of the channel. However, the mobile channel has unique characteristics that can result in several new security vulnerabilities.

Authentication is the remedy to the other attacks. User Authentication is defined as Provision of Assurance that the message is originated from authorized user (William). Message Authentication is defined as 'Provision of assurance that the message is not altered'. One type of Message Authentication is by hash algorithm. This provides an assurance to the destination that the message is not changed by the intruders (Nist, 2002). To get protected from eavesdropping, encryption in source and decryption in destination is also done with the help of a secret key. Encryption can be used to ensure secrecy, but other techniques are still needed to make communications secure, particularly to verify the integrity and authenticity of a message; for example, a Message Authentication Code (MAC) or digital signatures. Another consideration is protection against traffic analysis.

PROJECTED DESIGN

This proposal describes an proficient 256-bit hash function, GCSEC. It is designed not only to have higher seclusion but also efficiency. The recital of the new hash function is better than that of other algorithms in software.

For an idyllic hash utility with an m-bit output, finding a preimage or a second preimage requires about 2^m operations and the best ever way to unearth a conflict is a birthday attack which desires approximately $2^{m/2}$ operations. Most dedicated hash functions which have iterative procedure use the Merkle-Damgard edifice in order to hash inputs of arbitrary length. They work as follows. Let HASH be a hash function. The message X is padded to a multiple of the block length and subsequently divided into t blocks X1, • • •, Xt. Then HASH can be described as follows:

$$CV0 = IV; CVi = COMP(CVi\text{-}1, Xi),\ 1 \le i \le t;$$
$$HASH(X) = CVt,$$

Where, COMP is the solidity utility of HASH, CVi is the chaining capricious between stage i and stage i + 1 andIV denotes the preliminary estimation. The most trendy method of deceitful compression functions of obsessive hash functions is a serial successive iteration of a small step function, as like round functions of chunk ciphers. (Biham and Chen, 2004) specifies that many hash functions such as MD4, MD5, HAVAL, SHA-family,

Corresponding Author: M. Ganaga Durga, 119 Aruthu Pandiyar Street, Vasuki Nagar, Iii Street, Valluvar Colony, Madurai 625 017, Tamil Nadu, India

follow this. Attacks on hash functions have been paying attention on vanishing the difference of intermediate values caused by the difference of messages. MD4-type hash functions including SHA-1 are vulnerable to Wang collision-finding attack. RIPEMD-family has somewhat diverse approach for designing a secure hash function. The invader who tries to break members of RIPEMD family should aim simultaneously at two ways where the message difference passes. This design strategy is still successful because so far there is not any effective attack on RIPEMD-family except the first proposal of RIPEMD. However, Bellovin (1989) says that RIPEMD-family have heavier compression functions than hash functions with serial structure. Total number of steps is twice as many as that of MD4. Also, the number of steps of RIPEMD-160 is almost twice as many as that of SHA-0. In this study, we propose a dedicated hash function. According to the observation, we determined the design goals (of compression function) as follows:

- It should have a 256-bit output because the security of 2^{128} operations is recommended for symmetric key cryptography as the computing power increases.
- Its structure should be resistant against known attacks including Wang attack.
- The performance should be as competitive as that of SHA-256.

COMPUTATIONAL STEPS OF GCSEC

These are basic notations used in GCSEC:
+ : Addition mod 2^{32}
$\oplus$: XOR (exclusive OR)
<<<s : S-bit left rotation for a 32-bit string A

Input block length and padding: An input message is processed by 512-bit block. GCSEC pads a message by appending a single bit 1 next to the least significant bit of the message, followed by zero or more bit 0s until the length of the message is 448 modulo 512 andthen appends to the message the 64-bit original message length modulo 2^{64}.

Structure of GCSEC: The solidity function of GCSEC hashes a 512-bit string to a 256-bit string. It consists of five parallel branch functions, BRANCH1, BRANCH2, BRANCH3, BRANCH4 and BRANCH5 (Fig. 1). Let Cvi = (A, B, C, D, E, F, G, H) be the chaining variable of the density function. It is initialized to IV0 which is A = 6a09e667xB = bb67ae85xC = 3c6ef372xD = a54ff53axE = 510e527fxF = 9b05688cxG = 1f83d9abxH = 5be0cd19x.

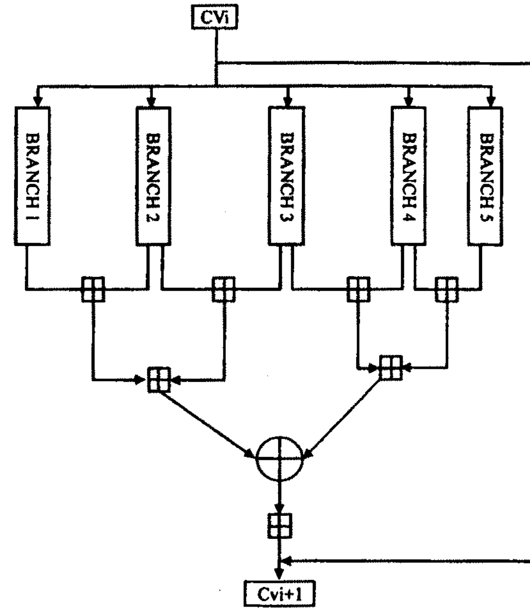


Fig. 1: Structure of GCSEC

Each consecutive 512-bit message block M is divided into sixteen 32-bit words M0, M1, • • •, M15 and the following computation is performed to update CVi to CVi+1:

Z = [BRANCH1(CVi, Σ_1(M)) + BRANCH2(CVi, Σ_2(M))]
Y = [BRANCH2(CVi, Σ_3(M)) + BRANCH3(CVi, Σ_2(M))]
X = [BRANCH3(CVi, Σ_3(M)) + BRANCH4(CVi, Σ_4(M))]
X1 = [BRANCH4(CVi,Σ_3(M)) + BRANCH5(CVi, Σ_4(M))]
X2 = Z+ Y and X3 = X+X1 and CVi+1 = CVi + [X2$\oplus$X3]

Where, Σ_j(M) = (Mσj(0), • • •, Mσj(15)) is the re-ordering of message words for j = 1, 2, 3, 4,5 given by Table 1.

ALGORITHM GUSH

Branch functions: Each BRANCHj is computed as follows:

- The chaining variable Cvi is copied to initial variables Vj, 0 for j-th branch.
- At k-th step of each BRANCHj ($0 \le k \le 7$), the step function STEPj, k is computed as follows:

Vj, k+1 = STEPj, k (Vj, k, M_j (2k), M_j (2k+1), αj, k, βj, k), where αj, k and βj, k are constants. Input Order of Message Words this table shows the input order of message words 0 to M 15 applied to BRANCHj ($1 \le j \le 5$) functions (Fig. 2).

Table 1: Ordering rule of message words

T	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
$\Sigma_1(t)$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
$\Sigma_2(t)$	15	16	12	10	9	11	4	5	3	14	1	6	7	8	13	2
$\Sigma_3(t)$	8	7	11	15	14	3	10	13	12	5	16	9	6	1	2	4
$\Sigma_4(t)$	6	13	2	9	16	1	14	12	4	11	10	3	8	15	5	7
$\Sigma_5(t)$	13	14	1	2	7	8	5	6	11	12	9	10	3	4	15	16

Table 2: Ordering of constants

Step k	α_{1k}	β_{1k}	α_{2k}	β_{2k}	α_{3k}	β_{3k}	α_{4k}	β_{4k}
0	δ_0	δ_1	δ_{15}	δ_{14}	δ_1	δ_0	δ_{14}	δ_{15}
1	δ_2	δ_3	δ_{13}	δ_{12}	δ_3	δ_2	δ_{12}	δ_{13}
2	δ_4	δ_5	δ_{11}	δ_{10}	δ_5	δ_4	δ_{10}	δ_{11}
3	δ_6	δ_7	δ_9	δ_8	δ_7	δ_6	δ_8	δ_9
4	δ_8	δ_9	δ_7	δ_6	δ_9	δ_8	δ_6	δ_7
5	δ_{10}	δ_{11}	δ_5	δ_4	δ_{11}	δ_{10}	δ_4	δ_5
6	δ_{12}	δ_{13}	δ_3	δ_2	δ_{13}	δ_{12}	δ_2	δ_3
7	δ_{14}	δ_{15}	δ_1	δ_0	δ_{15}	δ_{14}	δ_0	δ_1

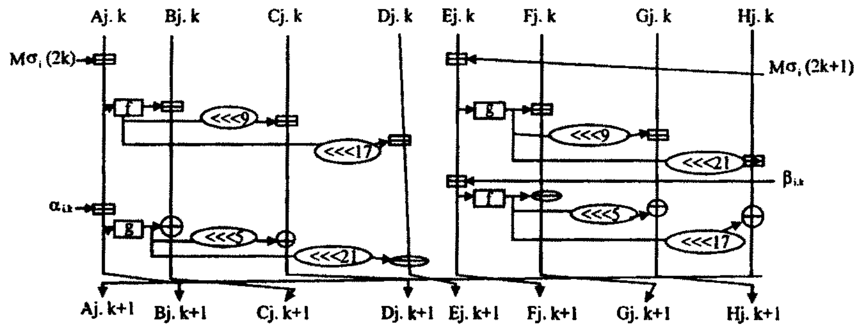


Fig. 2: The compression function of GCSEC

Branch computation: The compression function of GCSEC sixteen constants given by the following Table 2:

$\delta_0 = 428a2f98_x$	$\delta_1 = 71374491_x$
$\delta_2 = b5c0fbcf_x$	$\delta_3 = e9b5dba5_x$
$\delta_4 = 3956c25b_x$	$\delta_5 = 59f111f1_x$
$\delta_6 = 923f82a4_x$	$\delta_7 = ab1c5ed5_x$
$\delta_8 = d807aa98_x$	$\delta_9 = 12835b01_x$
$\delta_{10} = 243185be_x$	$\delta_{11} = 550c7dc3_x$
$\delta_{12} = 72be5d74_x$	$\delta_{13} = 80deb1fe_x$
$\delta_{14} = 9bdc06a7_x$	$\delta_{15} = c19bf174_x$

These constants are applied to each BRANCHj according to the ordering rule of them as follows:

f and g are nonlinear functions (https://www.cingular.com/media/text) as follows:

$$f(x) = x + (x<<<7 \oplus x<<<22),$$
$$g(x) = x \oplus (x<<<13 + x<<<27).$$

DEVISE THEORY

Structure GCSEC consists of 5 Branches. In the security facet, we can give the security against known attacks with the different message-ordering in branches. For example, RIPEMD, which consists of 2 branches, was fully attacked by Wang because RIPEMD has same message-ordering in 2 branches. On the other hand, in case of RIPEMD-128/160, there is no attack result because RIPEMD-128/160 have different message-ordering in branches. In the implementation aspect, GCSEC can be implemented efficiently because the message-ordering is simpler than the message expansion such as that of SHA-256 (Cellular Online, 2004) (Fig. 3).

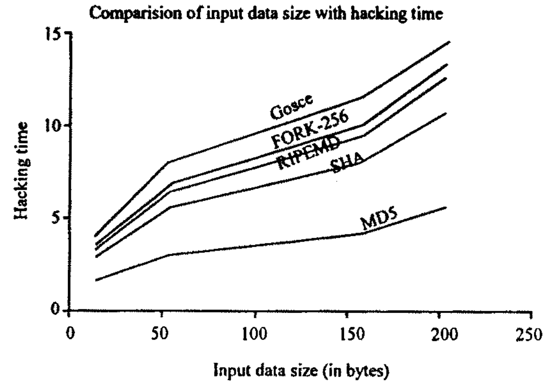


Fig. 3: Relationship of hacking speed

Constants: Each BRANCHi uses 16 different constants ái,j and âi,j for j =0, • • •, 7. By using constants we pursue the goal to disturb the attacker who tries to find a good differential characteristic with a relatively high probability.

Nonlinear functions: Nonlinear functions f and g output one word with one input word. Almost dedicated hash functions use boolean functions which output one word with three words at least. The boolean functions make it easy to control the output one word by adjusting the input several words. In addition, the output words of f and g functions are used to update other chaining variables. In almost dedicated hash functions output words of boolean functions are used to update only one chaining variable (CERT, 1996).

Ordering of message words: We espouse the message word ordering instead of the message word extension. If an attacker constructs an intended differential characteristics for one branch function, the ordering of message words will cause unintended differential patterns in the other branch functions (Cingubar Wireless). This is the core part of the security in the compression function.

GCSEC in hacking:

- Preimage and second preimage resistance is 2^{256}
- Birthday attack needs 2^{128}
- Probabilty to have Collision resistance is 2^{-256}
- Parallel operation is used to enhance the refuge.

SIMULATION

The hash algorithms effectively encrypt the messages and send them through mobile (CERT, 1996). The algorithm is developed in C and analyzed in MATLAB 6.1 for computers and mobile networks to encrypt and encrypt and decrypt the E-mail, Fax, SMS (Cisco Systems White Paper, 2004) and other real time data (Fig. 4).

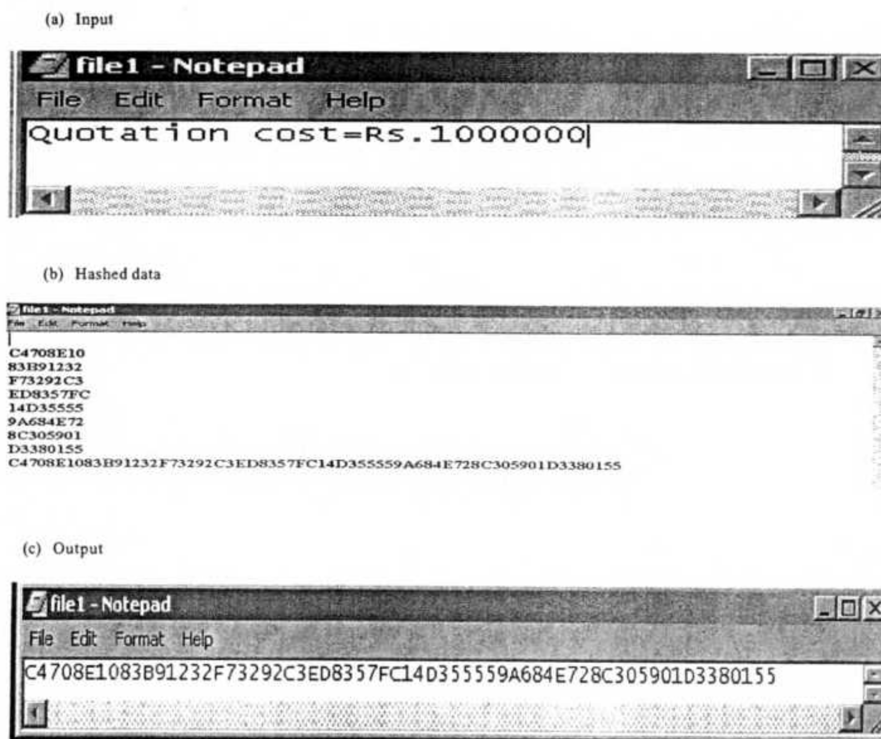


Fig. 4: The hash algorithms

CONCLUSION

All algorithms are compared with GCSEC algorithm. This algorithm works faster than others. The complexity of this algorithm is increased by introducing so many number of steps for shifting the left and right parts of the bits of the data. GCSEC is having highest level of privacy hence any hackers can't easily attack the non real time data. This algorithm can be used for sending or receiving non real time data in wireless or mobile networks.

ACKNOWLEDGMENT

The authors express their sincere thanks to the Principals of K.L.N. College of Engineering and MEPCO Schlenk Engineering College for their co-operation and constant encouragement.

REFERENCES

Biham, E. and R. Chen, 2004. Near Collisions of SHA-0. Advances in Cryptology CRYPTO 2004, LNCS 3152, Spriger-Verlag, pp: 290-305.

Bellovin, S., 1989. Security problems in the TCP/IP protocol suite. Comput. Commun. Rev., 19: 32-48.

Cellular Online, 2004. Uk sms traffic continues to rise. http://www.cellular.co.news.

CERT. Advisory CA-1996-26 'denial-of-service attack via ping'.http://www.cert.org/advisories/CA.

Cisco Systems Whitepaper, 2004. A study in mobile messaging: The evolution of messaging in mobile networks andhow to efficiently and effectively manage the growing messaging traffic. Technical report.

NIST/NSA, 2002. FIPS 180-2: Secure Hash Standard (SHS).

Rivest, R.L., 2004. The MD5 Message Digest Algorithm. IETF Request for Comments.

"CIPHER TALK-Encryption algorithm for mobile networks", Published in the Proceedings of the International Conference on Computational Intelligence and Multimedia Applications (ICCIMA 2007), Vol.4 , pp:363-367 and also published in IEEEXplore, December 2007.

Proceedings

International Conference on Computational Intelligence and Multimedia Applications

ICCIMA 2007

Volume – IV

13-15 December 2007
Sivakasi, Tamil Nadv, India



Los Alamitos, California
Washington • Tokyo



CIPHER TALK – Encrypted Instant Messaging In Mobile Networks

M.Ganaga durga,
Asst.Prof.,
K.L.N.College of Engg.
E-mail :
mgdurga@yahoo.com

Dr.G.Chandrasekaran,
Professor and Head,
M.C.A. dept.
Mepco Schlenk
Engg.college

Dr.S.Arivazhan
Professor and Head,
ECE dept.,
Mepco Schlenk
Engg.college

Abstract

Security requirements in any transaction are directly proportional to the transaction's value, sensitivity and volume. Encryption can be used to ensure secrecy, but other techniques are still needed to make communications secure, particularly to verify the integrity and authenticity of a message.

This manuscript describes an encryption algorithm for 3G mobiles and any type of networks for launching either online or offline messages. This algorithm receives the full message what is entered and it will be renewed into other format. Then the encrypted mobile data will be sent or received at the relevant places. This algorithm reduces the transmission delay and time by converting the message at their source itself and the decryption is done at the relevant place.

Keywords: ciphers,3G mobiles,voip,chat messages,hash

0-7695-3050-8/07 $25.00 © 2007 IEEE
DOI 10.1109/ICCIMA.2007.366

1.Introduction

Voice over IP and Instant Messaging are increasingly popular web-based communications systems for private, corporate, and academic purposes which do not have privacy to send or receive. Instant messengers can also provide an access point for backdoor trojan horses[2].

1.1. Securing instant messages

Securing instant messaging is not an easy task. One of the best ways to secure the information being transmitted along an instant messaging network is to encrypt it[1]. There are currently several companies that offer encrypted instant messaging communication. There are also instant messaging clients available that are compatible with some of the major networks that apply encryption to the instant messaging traffic. The proposed scheme gives an encryption algorithm for any type of data to maintain secrecy.

2.Maneuver

The chatting or instant messages like Net meeting can be encrypted using the hash algorithm of Gcsec. This algorithm first receives all the message from the source and send to the recipient after encrypting them in its originator itself. The originator and the receiver should have the private key to encrypt or decrypt their messages respectively in their mobiles as options which is to be motivated through the settings.

An key in communication is processed by 512-bit block . GCSec[6,7,8] hashes a 512-bit string to a 256-bit string using five parallel branch functions, BRANCH1,BRANCH2,BRANCH3,BRANCH4andBRANCH5. Let CVi = (A,B,C,D,E, F, G,H) be the chaining variable of the density function. It is initialized to IV0 which is:A = 6a09e667x B = bb67ae85x C = 3c6ef372x D = a54ff53ax E = 510e527fx F = 9b05688cx G = 1f83d9abx H = 5be0cd19xEach consecutive 512-bit message block M is divided into sixteen 32-bit words M0,M1, · · · ,M15 which are randomly selected and the following computation is performed to update CVi to CVi+1:

$Z=[BRANCH1(CVi, \sum_1(M)) + BRANCH2(CVi, \sum_2(M))]$

$Y= [BRANCH2(CVi, \sum_2(M)) + BRANCH3(CVi, \sum 3(M))]$

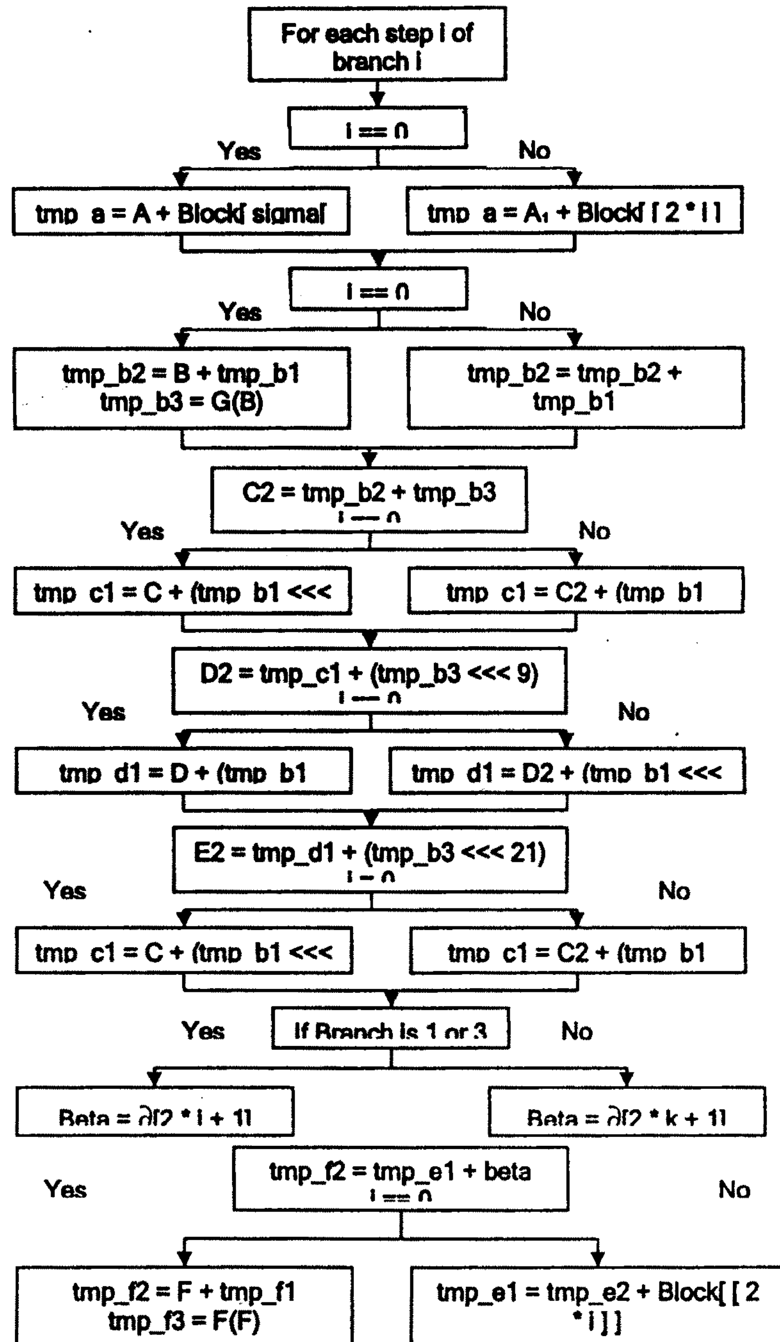
$X= [BRANCH3(CVi, \sum_3(M)) + BRANCH4(CVi, \sum_4(M))]$

$X1=[BRANCH4(CVi,\sum_4(M)) + BRANCH5(CVi, \sum 5(M))]$

$X2= Z+ Y$ & $X3= X+X1$ & $CVi+1 = CVi + [X2 \oplus X3]$

This algorithm hashes the unique ID or key to access the transactions can be given inturn as key to any data encryption algorithm like AES/DES to encrypt & decrypt the data which is to send or receive.

2.1.Algorithm steps:



3. Simulation

3.1. Carbide.j:

Carbide.j provides tools for creating and packaging Mobile Information Device Profile (MIDP) applications, and also provides a convenient interface for managing Nokia Java SDKs. The MIDP executable is a test environment that provides the cell phone image, display area, and key input[4,5]. Once an application has been created, the Nokia Developer's Suite for J2ME's SDK management features allow the application to be launched in multiple emulators for testing and verification[3].We have to specify the recipient name and phone number before typing the message in the chat room which indicates a new local chat room created for the new receipent Then the fig.1 shows that the packet of data sent which contains the address information of the recipient, the encrypted data and an address header identifying you to the recipient. The chart.1 & table.1 show the relationship between encryption times for various types of processors and mobiles.

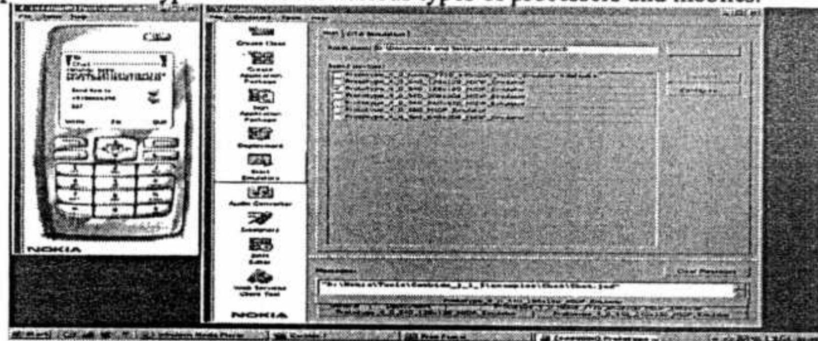


Figure 1.Encrypted message

Table 1.Encryption time relationships

DEVICES	TIME (SECS)
Nokia N70	1.4327
Nokia 7710	1.6736
Intel Dual Core	0.0121
Intel Hyper Threading	0.0196
Laptops	0.60652
Pentium IV	0.000435
486 DX100	0.00248
Pentium III	0.000065

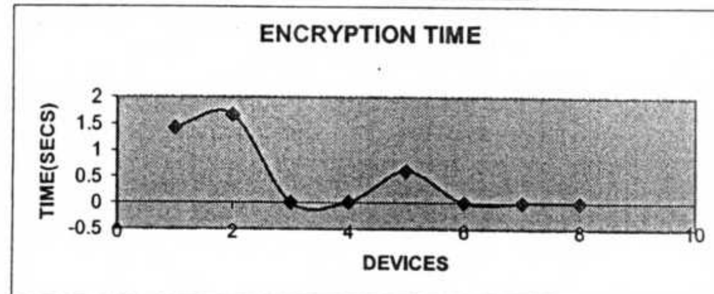


Chart 1.Encryption time relationships

The fig.3 shows the time taken to hash the key by all hashing algorithms.

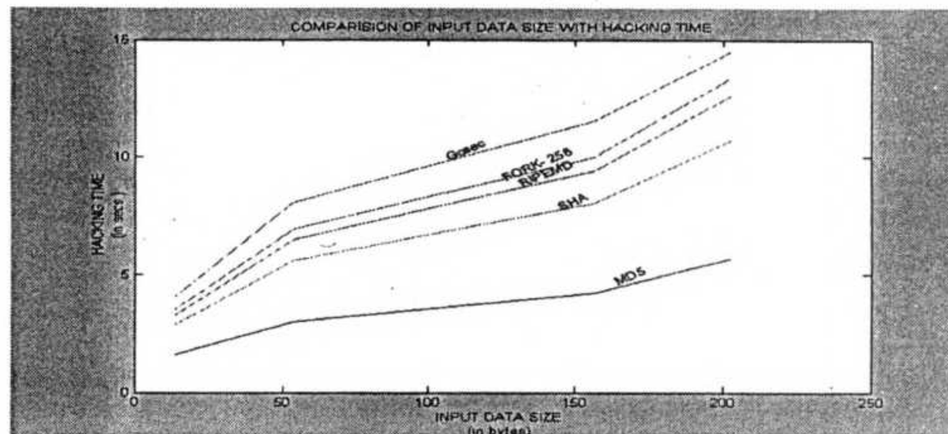


Figure 3.Hashing time relationships

4.Conclusion

The complexity of this algorithm is increased by introducing many number of steps for shifting the left and right parts of the bits of the data . Gcsec is having highest level of privacy since no hackers can attack the encrypted data. This algorithm is simulated for sending or receiving non real time data or real time in mobiles of S60 series like NOKIA 7710,N70 and various systems processors. This algorithm works faster than others with less amount of time and transmission delay because the conversion of data is done at the respective places itself. In future, the algorithm can be extended by repeating the inner looping operations of encryption with 'n' number of branches.

5.Acknowledgement

The authors express their sincere thanks to the Principals of K.L.N. College of Engineering & MEPCO Schlenk Engineering College for their co-operation and constant encouragement.

6.References

[1]Cingular,Wireless.Textmessaging,https://www.cingular.com/media/textmessaging purchase.

[2]Peterson, j. http://www.ietf.org/internet-drafts/draft-petersonmessage-identity-00.%txt, Oct. 2004.

[3]Campbell,B.Rosenberg,J schulzrinne, h.,huitema, c., and gurle, d. rfc 3428 – session initiation Protocol extension for instant messaging.http://www.ietf.org/rfc/rfc3428.txt, Dec.2002.

[4] ROSENBERG, J. RFC 3856 - a presence event package for the session initiation protocol (SIP).http://www.ietf.org/rfc/rfc3856.txt, Aug. 2004

[5] Saint-andre, P. RFC3920-3923 – IETF XMPP RFCs . http://www.ietf.org, Oct. 2004.

[6] M.Ganaga durga,Dr..G.Chandrasekaran, Dr.S.Arivazhagan Dedicated 256 hash algorithm for non real time data, Published in the Proceedings of the International Conference on communication & power systems(ICCPS'06) ,pp-198-202,2006.

[7]M.Ganaga durga, Dr.G.Chandrasekaran,Dr.S.Arivazhagan Hash Cryptography for SMS banking, Proceedings of the International Conference on Advances in Electronics & communications (icon ADELCO 2007), pp-133-138,2007.

[8]M.Ganaga durga, Dr.G.Chandrasekaran, Dr.S.Arivazhagan,GCSEC algorithm for Non real time data, Under publication of International journal of mobile communication..